PROGRAMAS DE JOGOS ESPACIAIS

Para o TK82, TK83, TK85, CP200, Apple, TRS 80

4ª EDIÇÃO

EDITORA LUTÉCIA

PROGRAMAS DE JOGOS ESPACIAIS

Para o TK82, TK83, TK85, CP200, Apple, TRS 80

Daniel Isaaman e Jenny Tyler

Tradução e Adaptação de
Ronaldo Sergio de Biasi

4ª EDIÇÃO

Sumário

Ilustrações
Martin Newton, Tony Baskeyfield, Graham Round,
Jim Bamber, Mark Duffin e John Bolton
Diagramação
Graham Round e Roger Priddy
Programa "Decolagem": Richard Nash
Programa "Os Olhos dos Monstros": Bob Merrv

LUTÉCIA

Introdução

Este livro contém programas de jogos espaciais para microcomputadores. Todos os programas são compatíveis com os micros fabricados no Brasil.

Embora os micros nacionais utilizem a linguagem BASIC, existem três dialetos ou variações dessa linguagem, dependendo da marca e modelo do micro. Os micros nacionais são divididos em três "famílias": a família Sinclair (TK-82C, TK-83, TK-85), a família TRS-80 (CP-300, DGT-1000, etc.) e a família Apple (Maxxi, Ap-II, etc.). Se você não souber a que família pertence o seu micro, poderá obter a informação em qualquer loja especializada.

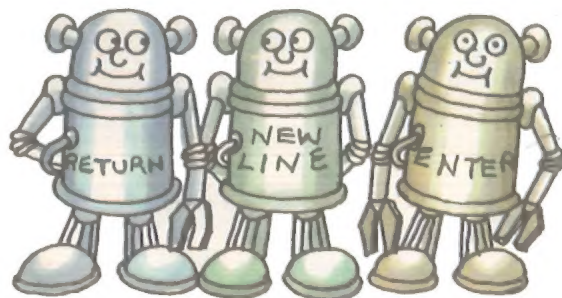
Neste livro, a listagem principal de cada programa é para os micros TK-82C e TK-83 (daqui por diante, tudo que dissermos do TK-83 também se aplica ao TK-82C). As linhas que precisam ser mudadas para que o programa funcione em outros computadores estão indicadas por símbolos e as modificações aparecem no final de cada listagem.

Para cada jogo, há idéias para modificações e extensões dos programas. No final do livro, você encontrará algumas "dicas" para escrever seus próprios programas. Incluímos também uma tabela de equivalência para ajudá-lo a adaptar ao seu micro programas publicados em revistas e em outros livros, além de uma lista das instruções de BASIC usadas neste livro, com as respectivas definições.

Como carregar os programas

Se o seu micro não é o TK-83, as linhas que devem ser mudadas estão indicadas pelos seguintes símbolos:

▲ TK-85
■ família TRS-80
● família Apple

Cada vez que você vir o símbolo da família a que pertence o seu micro, procure no final da listagem uma linha

com o mesmo símbolo e o mesmo número. Use-a no lugar da linha correspondente da listagem principal.

Pontos importantes

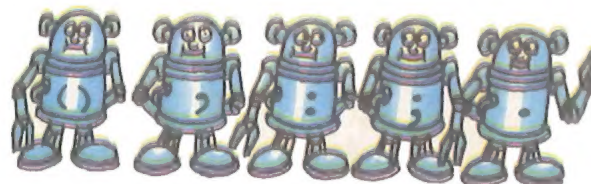
1 Escreva as linhas exatamente como na listagem, incluindo todos os espaços e sinais de pontuação.
2 Depois de bater cada linha, verifique na tela se está tudo certo.
3 Antes de começar uma nova linha, aperte a tecla RETURN, ENTER ou NEWLINE.
4 Cuidado para não pular uma linha ou misturar duas linhas. Use uma régua ou pedaço de papel para marcar o lugar onde está na listagem.
5 Preste atenção nos símbolos que identificam as diferentes famílias e use a instrução apropriada para o seu modelo de micro.
6 Se você está usando um micro TK-83 ou TK-85, não deve escrever as instruções letra por letra. Em vez disso, use as teclas correspondentes a cada instrução.

Talvez você prefira que alguém leia o programa para você enquanto aperta as teclas. Nesse caso, explique para o seu amigo que ele deve ler todas as vírgulas, pontos, parênteses e espaços.

Como verificar os programas

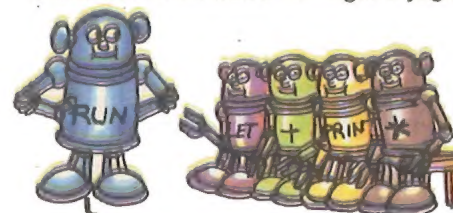
Depois de carregar o programa, consulte o manual do micro para ver como fazê-lo aparecer na tela. (Em geral, basta escrever LIST.) Verifique se todas as linhas estão corretas. É

muito fácil cometer erros; não se admire se encontrar alguns. Consulte o manual para saber como alterar as letras ou símbolos que estão errados. Em muitos casos, é mais fácil tornar a escrever a linha inteira. A linha antiga (com o mesmo número) é automaticamente apagada.

Eis uma lista dos erros mais comuns:
1 Linha faltando.
2 Linha com número errado.
3 O começo de uma linha emendado no final da anterior.
4 Falta de parênteses, vírgula, dois pontos, ponto e vírgula, ponto ou

espaço, especialmente nas linhas mais compridas. Cuidado com os parênteses duplos.
5 Linha errada para o seu modelo de micro.
6 Troca de zero pela letra "O".
7 Números errados (com zeros a mais, por exemplo).

Experiências com os jogos

Este livro contém várias sugestões para modificar e ampliar os programas, mas não tenha medo de introduzir suas próprias alterações. Não há perigo de danificar o computador; se sua idéia não der certo, você pode sempre voltar ao programa inicial.

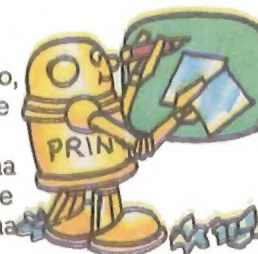
Você provavelmente terá vontade de mudar a velocidade de alguns jogos*, especialmente depois que estiver familiarizado com eles. As instruções para isso estão na página de cada programa.

Como executar os programas

Para começar qualquer jogo, você deve escrever RUN. Em alguns jogos, as coisas acontecem muito depressa, de modo que é melhor ler atentamente as instruções antes de começar.

É bastante provável que o programa ainda contenha alguns erros. Nesse caso, não acontecerá nada ou o jogo dará resultados imprevistos. Pode ser que o computador mostre na tela um código de erro, que você pode procurar no manual. Isso o ajudará a localizar o erro. Liste o programa de novo e compare as instruções com as do livro, uma a uma.

Quando o jogo terminar, aparecerá na tela uma mensagem (BREAK IN LINE 200, por exemplo). Para jogar de novo, você deve escrever RUN.

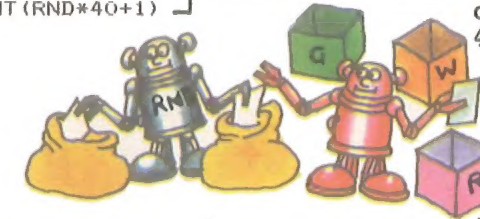
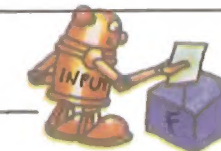
Você pode mudar à vontade o texto entre aspas que aparece depois de qualquer instrução PRINT. Também pode acrescentar novas mensagens. Para isso, escreva o número de uma linha que não aparece no programa (105, por exemplo, se você quer incluir uma mensagem entre as linhas 100 e 110), escreva PRINT e depois o texto da sua mensagem entre aspas.

Se o seu computador é capaz de gerar cores ou sons, consulte o manual para descobrir como funcionam as instruções correspondentes e experimente usá-las nos programas deste livro.

*Na página 35 há uma nota especial para os usuários do TK-85.

Decolagem

Você é o comandante de uma espaçonave. Sua nave caiu em um planeta desconhecido e você precisa decolar rapidamente em uma nave alienígena que conseguiu capturar. O computador da nave lhe diz qual é a força de gravidade do planeta, mas você tem que adivinhar qual a força que deve imprimir aos motores para decolar. Se a força for muito pequena, a nave não conseguirá subir. Se for muito grande, um mecanismo de segurança entrará em ação, desligando os motores para evitar que fiquem superaquecidos. Se ainda estiver no planeta depois de dez tentativas, será capturado pelos alienígenas.

Como o programa funciona

```
●10 CLS                              — Limpa a tela.
 20 PRINT "DECOLAGÉM"
■●30 LET G=INT(RND*20+1)             — O computador escolhe dois números: um entre 1 e 20, para
■●40 LET W=INT(RND*40+1)               colocar em G, e outro entre 1 e 40, para colocar em W.
 50 LET R=G*W                        — Multiplica o número em G pelo número em W. Coloca o
                                       resultado em R.
 60 PRINT "GRAVIDADE ";G             — Imprime GRAVIDADE e o número que está em G.
 70 PRINT "DIGA A FORCA"             — Pede a você para dar um palpite.
 80 FOR C=1 TO 10                    — Começa um "loop", dizendo ao computador para repetir as
                                       instruções seguintes 10 vezes.
 90 INPUT F                          — Guarda seu palpite em F.
 100 IF F>R THEN PRINT "MUITO GRANDE";   — Compara o número que está em F com o número que
 110 IF F<R THEN PRINT "MUITO PEQUENA";    está em R e imprime a mensagem apropriada ou pula
 120 IF F=R THEN GOTO 190                  para 190.
 130 IF C<>10 THEN PRINT ", TENTE DE NOVO" — Pede para você tentar de novo.
 140 NEXT C                          — Fim do "loop". Volta para a instrução 80.
 150 PRINT                           — Chega aqui depois de dez palpites errados.
 160 PRINT "VOCE FALHOU"
 170 PRINT "E FOI CAPTURADO"
 180 STOP
 190 PRINT "BOA VIAGEM"
 200 STOP
```

A listagem acima é para o TK-83. Para os outros computadores, faça as mudanças abaixo.

```
●10 HOME
●30 LET G=INT(RND(1)*20)
■30 LET G=INT(RND(0)*20)
●40 LET W=INT(RND(1)*40)
■40 LET W=INT(RND(0)*40)
```

Nos computadores tipo TRS-80 ou Apple, você pode usar END em lugar de STOP.

Como tornar o jogo mais difícil

Você pode reduzir o número de tentativas, mudando o último número da linha 80 e o número da linha 130. (Os dois números devem ser iguais.)

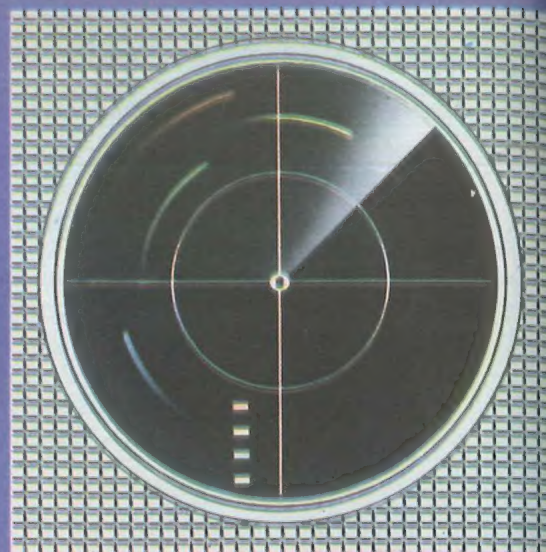
Problema

Como você faria para mudar a máxima força possível?

O significado de ■ ● está na pág. 2.

Jogos Intergalácticos

As redes de televisão da Terra estão competindo pela cobertura exclusiva dos Primeiros Jogos Intergalácticos. A companhia que conseguir primeiro colocar um satélite em órbita na altitude correta ganhará a concessão.

Você é o engenheiro que a Rede Século 21 encarregou de lançar o satélite. Será que vai conseguir lançá-lo com o ângulo e velocidade corretos?

Como o programa funciona

```
10 PRINT "JOGOS INTERGALACTICOS"
■●20 LET H=INT(RND*100+1)
30 PRINT "VOCE DEVE LANCAR UM SATELITE"
40 PRINT "ATE UMA ALTURA DE ";H
```
Escolhe a altitude, coloca em H e imprime.

```
50 FOR G=1 TO 8
```
Começa um "loop" de 8 tentativas.

```
60 PRINT "DIGA O ANGULO (0-90)"
70 INPUT A
```
Pede a você um ângulo e coloca em A.

```
80 PRINT "DIGA A VELOCIDADE (0-40000)"
90 INPUT V
```
Pede a você uma velocidade e coloca em V.

```
100 LET A=A-ATN(H/3)*180/3.14159
```
Usa H para calcular qual deve ser o ângulo e subtrai do seu palpite para ver se você chegou perto.

```
110 LET V=V-3000*SQR(H+1/H)
```
Calcula a velocidade correta e subtrai do seu palpite.

```
120 IF ABS(A)<2 AND ABS(V)<100 THEN GOTO 210
```
Verifica se você acertou e nesse caso pula para 210.

```
130 IF A<-2 THEN PRINT "MUITO BAIXO"
140 IF A>2 THEN PRINT "MUITO ALTO"
150 IF V<-100 THEN PRINT "MUITO LENTO"
160 IF V>100 THEN PRINT "MUITO RAPIDO"
```
Imprime um comentário para ajudá-lo na tentativa seguinte.

```
170 NEXT G
```
Volta para nova tentativa.

```
180 PRINT "VOCE FALHOU"
190 PRINT "ESTA DESPEDIDO"
```
Chega aqui depois de 8 palpites errados.

```
200 STOP
210 PRINT "VOCE CONSEGUIU"
220 PRINT "GANHAMOS...GRACAS A VOCE"
230 STOP
```
Chega aqui se você acertou.

A listagem acima é para o TK-83. Para os outros computadores, faça as mudanças abaixo.

```
■20 LET H=INT(RND(0)*100+1)
●20 LET H=INT(RND(1)*100+1)
```


Como aumentar o programa

Se você acrescentar as três linhas abaixo, o computador lhe dará um prêmio inversamente proporcional ao número de tentativas.

```
222 LET B=INT(1000/G)
225 PRINT "VOCE GANHOU UM"
227 PRINT "PREMIO DE ";B;" CREDITOS"
```

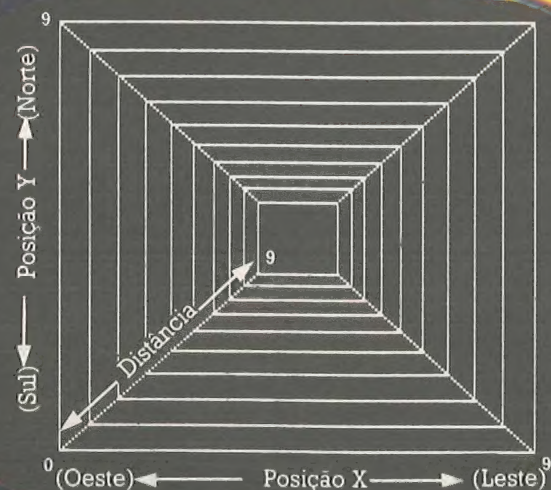
Problema

Como você mudaria o programa para que, se você ganhar, ele comece automaticamente outro jogo, somando o prêmio que você já ganhou aos novos prêmios?
(Pista: você precisa mudar duas linhas e acrescentar uma.)

Veja quantas vezes consegue jogar antes de ser despedido.

O significado de ■● está na pág. 2.

Pirata Cósmico

Em algum lugar atrás de você, na escuridão do espaço, está a nave de Elron, o Pirata Cósmico. Você sabe que ele está em algum ponto da rede tridimensional que aparece na tela da sua nave (figura abaixo). Você dispõe de quatro bombas. Pode lançá-las, uma a uma, em posições especificadas por três números entre 0 e 9, que deve fornecer ao computador da nave. Conseguirá acertar a nave de Elron antes que o pirata o capture?

Como o programa funciona

```
● 10 CLS
  20 PRINT "PIRATA COSMICO"
  30 LET S=10
  40 LET G=4
■● 50 LET X=INT(RND*S)
■● 60 LET Y=INT(RND*S)
■● 70 LET D=INT(RND*S)
  80 FOR I=1 TO G
  90 PRINT "POSICAO X (0 A 9)?"
  100 INPUT X1
  110 PRINT "POSICAO Y (0 A 9)?"
  120 INPUT Y1
  130 PRINT "DISTANCIA (0 A 9)?"
  140 INPUT D1
  150 IF X=X1 AND Y=Y1 AND D=D1 THEN GOTO 280
  160 PRINT "TIRO FOI PARA ";
  170 IF Y1>Y THEN PRINT "N";
  180 IF Y1<Y THEN PRINT "S";
  190 IF X1>X THEN PRINT "E";
  200 IF X1<X THEN PRINT "O";
  210 PRINT
  220 IF D1>D THEN PRINT "MUITO LONGE"
  230 IF D1<D THEN PRINT "MUITO PERTO"
  240 NEXT I
  250 PRINT
  260 PRINT "VOCE FOI CAPTURADO!!"
  270 STOP
  280 PRINT
  290 PRINT "*BUUUM* VOCE O PEGOU!!"
  300 STOP
```

- Linha 30: Fixa o tamanho da rede.
- Linha 40: Fixa o número de tentativas.
- Linhas 50–70: A posição de Elron é fixada por essas três linhas.
- Linha 80: Começo de um "loop" que diz ao computador para repetir as próximas 15 linhas G vezes.
- Linhas 90–140: Pede a você três números e coloca em X1, Y1 e D1.
- Linha 150: Verifica se você acertou e nesse caso pula para 300.
- Linhas 160–230: Compara seus palpites com a posição de Elron e dá uma pista.
- Linha 240: Fim do "loop". Volta para outra tentativa.
- Linha 260: Chega aqui depois de G palpites errados.
- Linha 290: Chega aqui se você acertou.

A listagem acima é para o TK-83. Para os outros computadores, faça as mudanças abaixo.

```
● 10 HOME
● 50 LET X=INT(RND(1)*S)
■ 50 LET X=INT(RND(0)*S)
● 60 LET Y=INT(RND(1)*S)
■ 60 LET Y=INT(RND(0)*S)
● 70 LET D=INT(RND(1)*S)
■ 70 LET D=INT(RND(0)*S)
```

Como tornar o jogo mais difícil

Este programa foi escrito de tal forma que você pode mudar facilmente a dificuldade alterando o tamanho da rede. Para isso, coloque um valor diferente de S na linha 30.

Se você aumentar o tamanho da rede, precisará de mais bombas para ter uma possibilidade razoável de acertar a nave de Elron. Para isso, mude o valor de G na linha 40.

Problema

Como você mudaria o programa para que o computador lhe peça o valor de um número de dificuldade para colocar em S? (Pista: limite o valor de S a um número entre 6 e 30 e use INT (S/3) para o valor de G na linha 40.)

O significado de ■ ● está na pág. 2.

Os Olhos dos Monstros

Você está encurralado! Para onde quer que se volte, os olhos frios e assustadores de um monstro do espaço o espreitam por um momento, antes que o animal se esconda atrás de uma pedra. Pouco a pouco, os monstros se aproximam de você, tentando envolvê-lo em seus tentáculos. Felizmente, você está com a sua pistola de prótons.

Os olhos dos monstros aparecem em quatro lugares diferentes da tela, que correspondem às teclas de 1 a 4. Se você apertar a tecla certa enquanto os olhos do monstro estiverem na tela, ele será destruído. Existem 10 monstros... à medida que você os destrói, sua chance de escapar aumenta.

Como o programa funciona

```
10 PRINT "OS OLHOS DOS MONSTROS"
20 LET S=0
```
Coloca a contagem em zero para começar o jogo.

```
30 FOR T=1 TO 10
```
Início de um "loop" para 10 monstros.

```
●40 CLS
```
Limpa a tela.

```
■●50 FOR I=1 TO INT(RND*30+20)
60 NEXT I
```
"Loop" de espera que dura um tempo variável, que depende do valor de RND.

```
■●70 LET R=INT(RND*4+1)
```
Escolhe um número de 1 a 4 e coloca em R.

```
■●80 GOSUB 210+30*R
```
Pula para uma das quatro sub-rotinas, dependendo do valor de R. Cada uma delas coloca o cursor em uma posição diferente da tela: "A" espaços na horizontal e "D" linhas na vertical.

```
90 PRINT "OO"
```
Coloca os "olhos do monstro" na tela.

```
■●100 FOR I=1 TO 20
●→
■●110 LET R$=INKEY$
120 IF R$<>"" THEN GOTO 140
130 NEXT I
140 IF VAL("0"+R$)<>R THEN GOTO 210
```
"Loop" para ver se você está apertando uma tecla. Se está, o computador pula para 140 e verifica se é a tecla certa.

```
150 LET S=S+1
```
Aumenta a contagem de 1.


```
●160 CLS
```
Limpa a tela.

```
170 GOSUB 350
180 PRINT "*"
```
Leva o cursor de volta para a mesma posição e coloca uma estrela na tecla.

```
■●190 FOR J=1 TO 40
200 NEXT J
```
"Loop" de espera para deixar a estrela um tempo suficiente na tela.

```
210 NEXT T
```
Volta para o monstro seguinte.

```
220 PRINT "VOCE MATOU ";S;"/10 MONSTROS"
```
Imprime a contagem final.

```
230 STOP
240 LET D=5
250 LET A=1
260 GOTO 350
270 LET D=1
280 LET A=9
290 GOTO 350
300 LET D=5
310 LET A=18
320 GOTO 350
330 LET D=10
340 LET A=7
350 FOR I=1 TO D
360 PRINT
370 NEXT I
380 PRINT TAB(A);
390 RETURN
```
Sub-rotinas.

ENTRADA DA SUB-ROTINA

GOSUB

SAÍDA DA SUB-ROTINA

RETURN

A listagem acima é para o TK-83. Para os outros computadores, faça as mudanças abaixo.

```
●40,160 HOME
■50 FOR I=1 TO INT(RND(0)*300+200)
●50 FOR I=1 TO INT(RND(1)*300+200)
■70 LET R=INT(RND(0)*4+1)
●70 LET R=INT(RND(1)*4+1)
■●80 ON R GOSUB 240,270,300,330
●100 FOR I=1 TO 150
●105 R$=""
●110 IF PEEK(-16384)>127 THEN GET R$
■●190 FOR J=1 TO 300
```

Como mudar a velocidade

Você pode acelerar o jogo diminuindo o último número da linha 100.

Como aproveitar melhor a tela

Você pode fazer com que os olhos apareçam em posições mais afastadas. Para isso, basta mudar os valores de A nas linhas 250, 280, 310 e 340. Consulte o manual do seu micro para ver o valor máximo que pode usar.

Problema

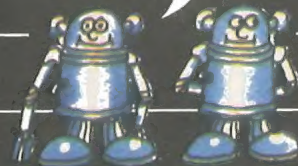
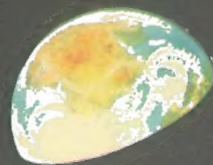
Como você mudaria o programa para os olhos aparecerem em mais de cinco pontos da tela? E para apareceram mais olhos?

O significado de ■ ● está na pág. 2.

Módulo Lunar

Você está no controle de um módulo lunar que está levando um pequeno grupo de astronautas para uma superfície da Lua. Para pousar com segurança, você precisa reduzir a velocidade disparando um retrofoguete, mas seu suprimento de combustível é limitado.

O computador de bordo lhe diz qual é a altitude inicial, a velocidade e o suprimento de combustível e lhe pergunta a quantidade de combustível que deseja queimar. Em seguida, calcula a nova altitude e velocidade. Queimando cinco unidades de combustível, você mantém a velocidade constante. Uma quantidade maior diminui a velocidade. Procure pousar com a menor velocidade possível. Será que *você* vai conseguir pousar na Lua?

Repare nas vírgulas e ponto e vírgula das linhas 70 e 80. Experimente omiti-las ou permutá-las para ver o que acontece.

```
● 10 CLS
  20 PRINT "MODULO LUNAR"
  30 LET T=0
  40 LET H=500
  50 LET V=50
  60 LET F=120
  70 PRINT "TEMPO";T,"ALTITUDE";H
  80 PRINT "VELOC";V,"COMBUSTIVEL";F
  90 IF F=0 THEN GOTO 140
 100 PRINT "QUEIMA? (0-30)"
 110 INPUT B
 120 IF B<0 THEN LET B=0
 130 IF B>30 THEN LET B=30
 140 IF B>F THEN LET B=F
 150 LET V1=V-B+5
 160 LET F=F-B
 170 IF (V1+V)/2>=H THEN GOTO 220
 180 LET H=H-(V1+V)/2
 190 LET T=T+1
 200 LET V=V1
 210 GOTO 70
 220 LET V1=V+(5-B)*H/V
 230 PRINT
 240 IF V1>5 THEN PRINT "VOCES CAIRAM...TODOS MORTOS"
 250 IF V1>1 AND V1<=5 THEN PRINT "OK...MAS ALGUNS FERIDOS"
 260 IF V1<=1 THEN PRINT "BOA ATERRISSAGEM"
 270 STOP
```

Como o programa funciona

- Linhas 10–80: Fixa e imprime os valores iniciais do tempo, altitude, velocidade e combustível.
- Linha 90: Se acabou o combustível, o computador pula a parte em que pergunta a você quanto combustível deve queimar.
- Linhas 100–140: Pergunta a você quanto combustível deve queimar e verifica se está dentro dos limites.
- Linha 150: Calcula a nova velocidade, V1.
- Linha 160: Calcula quanto combustível resta.
- Linha 170: Verifica se a distância percorrida desde a última queima é maior ou igual a altitude. Em caso afirmativo, você pousou. O computador então pula para a parte em que verifica se você pousou em segurança.
- Linha 180: Calcula a nova altitude.
- Linha 190: Aumenta o tempo de 1.
- Linha 200: Coloca a nova velocidade em V para que seja impressa na Linha 80.
- Linha 210: Volta ao começo do "loop".
- Linhas 220–270: Calcula a sua velocidade final e verifica se você pousou bem.

A listagem acima é para o TK-83. Para os outros computadores, faça a mudança abaixo.

```
● 10 HOME
```

O significado de ● está na pág. 2.

Como aumentar o programa

Se você acrescentar as linhas abaixo, aparecerá uma estrela nas posições sucessivas da nave. A distância da estrela até o lado esquerdo da tela corresponde à altitude da nave.

```
85 FOR I=2 TO H/500*nn
86 PRINT " ";
87 NEXT I
88 PRINT "*"
```

Substitua nn pela largura da sua tela.

Faça uma experiência

Experimente mudar os valores de H, V e F nas linhas 40 a 60 para ver o que acontece.

Problema

Você pode tornar o jogo mais fácil aumentando a velocidade máxima permitida para uma aterrissagem segura. Como você mudaria o programa para que isso acontecesse?

Feras de Galacticon

Pousar em Galacticon foi fácil... mas ninguém avisou a você que ali vivem algumas das mais perigosas feras do Universo.

Ao ser atacado por um dos monstros, você precisa escolher uma de suas armas — uma pistola de raios, um sabre de *laser* ou um triturador molecular — para usar contra ele. Se fizer a escolha correta, talvez consiga sobreviver para conquistar Galacticon.

Como o programa funciona

```
10 PRINT "FERAS DE GALACTICON"
20 DIM M$(4)
```

Reserva uma área de armazenamento (um "array") chamada M$ com 4 compartimentos: M$(1), M$(2), M$(3) e M$(4).

```
30 LET N=4
```

Fixa o número de monstros em 4.

```
40 LET M=5
```

Fixa o número de astronautas em 5.

```
50 LET M$(1)="SULFACIDOR"
60 LET M$(2)="FLAMGONDAR"
70 LET M$(3)="BALNOLOTIN"
80 LET M$(4)="GOLANDOR"
```

Coloca os nomes dos 4 monstros no "array".

```
    90 FOR I=1 TO N
■● 100 LET A=INT(RND*N+1)
■● 110 LET B=INT(RND*N+1)
   120 LET T$=M$(A)
   130 LET M$(A)=M$(B)
   140 LET M$(B)=T$
   150 NEXT I
```

Estas linhas misturam os nomes dos monstros. O computador executa esta parte N vezes. De cada vez, escolhe dois números entre 1 e N e troca os nomes que estão nos compartimentos com esses números. T$ é uma variável auxiliar usada no processo.

```
   160 FOR T=1 TO 8
```

Começo de um "loop" de oito ataques.

```
 ● 170 CLS
```

Limpa a tela.

```
■● 180 LET R=INT(RND*N+1)
   190 PRINT "MONSTRO CHEGANDO..."
   200 PRINT "E UM ";M$(R)
```

Escolhe um dos monstros e imprime seu nome.

```
    210 PRINT "QUE ARMA ? (R,S OU T) "
    220 INPUT R$
▲■● 230 LET W=CODE(R$)-54+R
  ■ 240 LET W=W-3*(W>3)-3*(W>6)
```

O computador usa os valores de R e R$ para calcular um número, W, que pode ser 1, 2 ou 3.

```
   250 IF W=2 THEN GOTO 300
```

Se W é 2, o computador pula para 300 para informar que você matou um monstro.

```
   260 IF W=3 THEN GOTO 320
```

Se W é 3, o computador pula para 320 para informar "sem efeito".

```
   270 PRINT "NAO ADIANTA. COMEU UM"
   280 PRINT "DO SEU GRUPO"
   290 GOTO 360
```

Chega aqui se W não é 2 nem 3. Depois, pula para 360 para diminuir de 1 o número de astronautas.

```
   300 PRINT "VOCE O MATOU"
   310 GOTO 380
   320 PRINT "SEM EFEITO"
■● 330 IF RND>.4 THEN GOTO 380
   340 PRINT "VOCE ENFURECEU O ";M$(R);"E ELE"
   350 PRINT "MATOU UM DO SEU GRUPO"
```

Se a arma não faz efeito, há uma certa probabilidade de que o monstro mate alguém.

```
   360 LET M=M-1
   370 IF M<1 THEN GOTO 440
```

Verifica se ainda resta algum astronauta. Se não, pula para 440.

```
■● 380 FOR I=1 TO 20
   390 NEXT I
```

"Loop" de espera para que você possa ler as mensagens na tela.

```
   400 NEXT T
```

Volta para o ataque seguinte.

```
   410 PRINT "VOCE SOBREVIVEU PARA"
   420 PRINT "CONQUISTAR GALACTICON"
   430 STOP
   440 PRINT "VOCES MORRERAM TODOS"
   450 STOP
```

Chega aqui se há sobreviventes depois de todos os ataques.

A listagem acima é para o TK-83. Para os outros computadores, faça as mudanças abaixo.

```
 ■ 100 LET A=INT(RND(0)*N+1)
 ● 100 LET A=INT(RND(1)*N+1)
 ■ 110 LET B=INT(RND(0)*N+1)
 ● 110 LET B=INT(RND(1)*N+1)
 ● 170 HOME
 ■ 180 LET R=INT(RND(0)*N+1)
 ● 180 LET R=INT(RND(1)*N+1)
■● 230 LET W=ASC(R$)-81+R
 ▲ 230 LET W=CODE(R$)-81+R
 ■ 240 LET W=W+3*(W>3)+3*(W>6)
 ● 330 IF RND(1)>.4 THEN GOTO 380
 ■ 330 IF RND(0)>.4 THEN GOTO 380
■● 380 FOR I=1 TO 300
```

Problema

Há pelo menos quatro maneiras de tornar este jogo mais difícil. Você sabe quais são?

O significado de ▲ ■ ● está na pág. 2.

Emboscada

Você é o comandante de um cruzador interestelar que, por causa de um defeito nos motores de hiperespaço, entrou em uma zona proibida. Sua nave está sendo atacada por caças alienígenas, que, para tornar as coisas piores, estão usando um aparelho de despistamento que faz com que o seu radar forneça leituras falsas. Felizmente, o computador de bordo fornece um número de código que você pode usar para calcular a posição correta das naves inimigas. Mas você precisa ser rápido ... elas não ficam muito tempo no mesmo lugar!

O computador imprime uma letra (a posição falsa do inimigo) e um número. Você deve avançar no alfabeto esse número de letras e apertar a tecla correspondente. Por exemplo: para M4 você deve apertar a tecla Q, para C2 a tecla E, e assim por diante. Quando você aperta uma tecla, dispara automaticamente o canhão de *laser*, e, se a letra estiver certa, destrói uma nave inimiga. Você pode escolher a dificuldade de cada jogo, escolhendo um número de 1 a 10. Este número corresponde ao maior número que pode aparecer no código. Em cada jogo, você é atacado por 10 caças alienígenas. Veja quantos consegue acertar.

O significado de ▲ ■ ● está na pág. 2.

Como o programa funciona

```
● 10 CLS
  20 PRINT "EMBOSCADA"
  30 PRINT
  40 PRINT "DIFICULDADE (1-10)"
  50 INPUT D
  60 IF D<1 OR D>10 THEN GOTO 50
```
Pede a você um número de dificuldade, coloca em D e verifica se está dentro dos limites.

```
  70 LET S=0
```
Inicia a contagem em zero.

```
  80 FOR G=1 TO 10
```
Começo de um "loop" de 10 tentativas.

```
▲■● 90 LET L$=CHR$(INT(RND*(26-D)+38))
```
Escolhe uma letra entre A e D letras antes de Z.

```
■● 100 LET N=INT(RND*D+1)
```
Escolhe um número entre 1 e D.

```
  110 CLS
  120 PRINT
  130 PRINT L$,N
```
Imprime a letra e o número.

```
■● 140 FOR I=1 TO 20+D*5
■● 150 LET I$=INKEY$
   160 IF I$<>"" THEN GOTO 190
   170 NEXT I
```
Verifica se você está apertando uma tecla e pula para 190 se você está.

```
  180 GOTO 200
```
Se você não apertou nenhuma tecla, pula para 200 e começa uma nova tentativa.


```
■● 190 IF I$=CHR$(CODE(L$)+N) THEN LET S=S+1
```
Verifica se você apertou a tecla certa; em caso afirmativo, aumenta a sua contagem de 1.

```
  200 NEXT G
```
Fim do "loop". Volta para outra tentativa.

```
  210 PRINT "VOCE ACERTOU ";S;"/10"
  220 STOP
```
Imprime a contagem depois de 10 tentativas.

A listagem acima é para o TK-83. Para os outros computadores faça as mudanças abaixo.

```
● 10,110 HOME
■ 90 LET L$=CHR$(INT(RND(0)*(26-D)+65))
● 90 LET L$=CHR$(INT(RND(1)*(26-D)+65))
▲ 90 LET L$=CHR$(INT(RND*(26-D)+65))
■ 100 LET N=INT(RND(0)*D+1)
● 100 LET N=INT(RND(1)*D+1)
■ 140 FOR I=1 TO 200+D*50
● 140 FOR I=1 TO 100+D*50
● 145 I$=""
● 150 IF PEEK(-16384)>127 THEN GET I$
■● 190 IF I$=CHR$(ASC(L$)+N) THEN LET S=S+1
```

Como mudar a velocidade

Se o jogo é rápido demais para você, coloque um número maior no meio da linha 140 (isto é, no lugar de 20, 100 ou 200). Você também pode aumentar a velocidade do jogo, usando um número menor.

Como tornar o jogo mais difícil

Você pode mudar o 1 nas linhas 40 e 60 para 3, por exemplo. Nesse caso, a dificuldade terá que ser no mínimo 3.

Problema

Mude a contagem para que corresponda ao número de código, isto é, para que você faça 1 ponto se o código for 1, 2 pontos se o código for 2, etc.

Cinturão de Asteróides

Você está atravessando o Cinturão de Asteróides. Para não se chocar com eles, precisa destruí-los; a força necessária para isso depende do tamanho do asteróide.

Os asteróides aparecem na tela do seu computador como grupos de estrelas. Para destruí-los, você precisa apertar a tecla do número correspondente ao número de estrelas. Não perca tempo... os asteróides são muito velozes!

Como o programa funciona

```
   10 PRINT "CINTURAO DE ASTEROIDES"
   20 LET S=0                          — Inicia a contagem em zero.
   30 FOR G=1 TO 10                    — Começa um "loop" de 10 asteróides.
●  40 CLS
■● 50 LET A=INT(RND*18+1)              — Escolhe um número para a posição horizontal do asteróide na tela e coloca em A.
■● 60 LET D=INT(RND*12+1)              — Escolhe um número para a posição vertical do asteróide na tela e coloca em D.
   70 LET N=INT(RND*9+1)               — Escolhe um número de 1 a 9 para o tamanho do asteróide.
   80 FOR I=1 TO D
   90 PRINT                            — Move o cursor D linhas para baixo.
  100 NEXT I
  110 FOR J=1 TO N
  120 IF J<>1 AND J<>4 AND J<>7 THEN GOTO 150
  130 PRINT
  140 PRINT TAB(A);                    — Coloca N estrelas na tela.
  150 PRINT "*";
  160 NEXT J
  170 PRINT
●→
■●180 FOR I=1 TO 10
● 190 LET Q=VAL("0"+INKEY$)            — Verifica se você está apertando uma tecla; se estiver, pula para 240.
  200 IF Q<>0 THEN GOTO 240
  210 NEXT I
  220 PRINT "BATEU NO ASTEROIDE"       — Chega aqui se o seu tempo acabou.
  230 GOTO 290
  240 IF Q<>N THEN GOTO 270            — Verifica se o seu número é igual a N; se não for, pula para 270.
  250 PRINT "ASTEROIDE DESTRUIDO"      — Chega aqui se você apertou o número certo.
  260 LET S=S+1                        — Aumenta a contagem de 1.
  270 IF Q<N THEN PRINT "MUITO FRACO"  — Compara o seu número com N e imprime a mensagem apropriada.
  280 IF Q>N THEN PRINT "MUITO FORTE"
■●290 FOR I=1 TO 50                    — "Loop" de espera para que você possa ler as mensagens.
  300 NEXT I
  310 NEXT G                           — Volta para mais um asteróide.
  320 PRINT "VOCE ACERTOU ";S;" EM 10" — Imprime a contagem depois de dez asteróides.
  330 STOP
```


A listagem acima é para o TK-83. Para os outros computadores, faça as mudanças abaixo.

```
●  40 HOME
■  50 LET A=INT(RND(0)*18+1)
●  50 LET A=INT(RND(1)*18+1)
■  60 LET D=INT(RND(0)*12+1)
●  60 LET D=INT(RND(1)*12+1)
■  70 LET N=INT(RND(0)*9+1)
●  70 LET N=INT(RND(1)*9+1)
● 175 Q=0
■●180 FOR I=1 TO 100
● 190 IF PEEK(-16384)>127 THEN GET Q
■ 290 FOR I=1 TO 500
● 290 FOR I=1 TO 250
```

Como mudar a velocidade do jogo

A linha 180 controla o tempo que você tem para apertar uma tecla. Para acelerar o jogo, diminua o último número da linha 180.

Problema

Você sabe como mudar o sistema de contagem para que o número de pontos seja igual ao número de estrelas de cada asteróide destruído?

O significado de ■● está na pág. 2.

Viagem ao Futuro

Imagine que você está em uma espaçonave viajando com uma velocidade próxima da luz. O tempo passa mais lentamente na espaçonave do que na Terra. Assim depois de uma longa viagem, você pode voltar à Terra em uma data posterior à indicada pelo relógio da nave. Neste jogo, o computador diz quantos anos deverão ter passado na Terra até a sua volta. Você então tem que escolher a distância a ser percorrida (em anos-luz) e a velocidade da nave (em relação à velocidade da luz) para chegar à Terra na data prevista. Cuidado para não viajar para muito longe e muito devagar, ou poderá morrer de velhice no caminho.

Como o programa funciona

```
●10 CLS
 20 PRINT "VIAGEM AO FUTURO"
■●30 LET T=INT(RND*100+25)
 40 PRINT "VOCE DEVE VOLTAR"
 50 PRINT "DAQUI A ";T;" ANOS."
 60 PRINT
```

Escolhe e imprime um número inteiro entre 25 e 124.

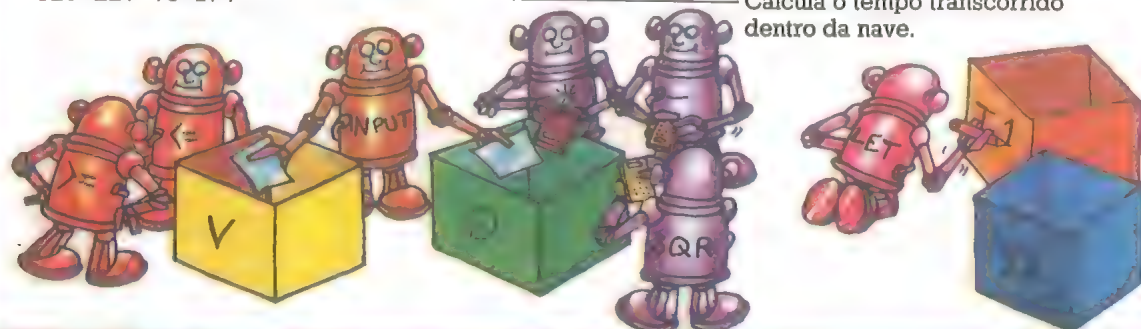
```
 70 PRINT "VELOCIDADE DA NAVE (0-1)"
 80 INPUT V
 90 IF V>=1 OR V<=0 THEN GOTO 70
```

Pede a você uma velocidade e verifica se está dentro dos limites.

```
100 PRINT "DISTANCIA A PERCORRER"
110 INPUT D
```

Pede a você uma distância.

```
120 LET T1=D/V
```

Calcula o tempo transcorrido dentro da nave.

```
130 LET T2=T1/SQR(1-V*V)
```

Calcula o tempo transcorrido na Terra.

```
140 PRINT "VOCE LEVOU ";T1;"ANOS"
150 PRINT "E CHEGOU DEPOIS DE"
160 PRINT T2;" ANOS."
```

Imprime os tempos.

```
170 IF T1>50 THEN GOTO 220
```

Verifica se você levou mais que 50 anos na viagem. Em caso afirmativo, pula para 220 e imprime uma mensagem.

```
180 IF ABS(T-T2)<=5 THEN PRINT "VOCE CHEGOU NA HORA"
190 IF (T-T2)>5 THEN PRINT "VOCE CHEGOU MUITO CEDO"
200 IF (T-T2)<-5 THEN PRINT "VOCE CHEGOU MUITO TARDE"
210 STOP
220 PRINT "VOCE MORREU NO CAMINHO"
230 STOP
```

Verifica se você chegou perto da data prevista e imprime uma mensagem apropriada.

A listagem acima é para o TK-83. Para os outros computadores, faça as mudanças abaixo.

```
●10 HOME
■30 LET T=INT(RND(0)*100+25)
●30 LET T=INT(RND(1)*100+25)
```

Problema

Mude o programa para:
1) Aumentar o número de anos que devem passar na Terra.
2) Diminuir a tolerância na sua data de chegada de 5 para 2 anos.
3) Aumentar (ou diminuir) o seu tempo de vida.

O significado de ■ ● está na pág. 2.

Vale da Morte

Só há um meio de escapar ao ataque dos perigosos Dissectitrons. Você vai ter que respirar fundo e entrar com a sua nave no perigoso desfiladeiro conhecido como Vale da Morte.

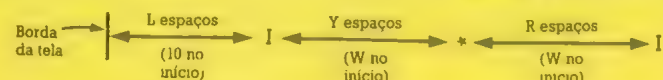
Seu computador vai querer saber a largura do vale. Comece com 15 e vá diminuindo aos poucos; 8 já é bastante difícil. Dirija sua nave apertando Q para ir para a esquerda e P para ir para a direita e veja se consegue atravessar o Vale da Morte.

Como o programa funciona

```
10 PRINT "VALE DA MORTE"
20 LET S=0
```
Fixa em zero sua posição no início do jogo.

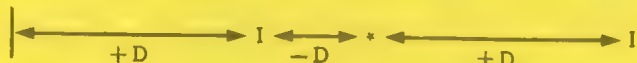
```
30 LET M=200
```
M é sua posição no final do jogo.

```
40 PRINT "LARGURA ?"
50 INPUT W
60 LET W=INT(W/2)
```
Pede a você a largura do vale, divide por 2 e usa INT para tirar os quebrados.

```
70 LET L=10
80 LET Y=W
90 LET R=W
```
Y e R são as distâncias entre as paredes e a nave.


```
■● 100 LET D=INT(RND*3-1)
```
Escolhe −1, 0 ou 1 e coloca em D.

```
110 IF L+D<0 OR L+D>20 THEN GOTO 100
```
Verifica se L+D não faz as paredes desaparecerem da tela.

```
120 LET L=L+D
130 LET Y=Y-D
140 LET R=R+D
```
Muda a posição das paredes de acordo com o valor de D.


```
▲■● 150 SCROLL
160 LET N=L
170 GOSUB 1000
180 PRINT "I";
```
Move o cursor L espaços na tela e imprime I. (O ponto e vírgula é para o cursor não mudar de linha.)

```
190 LET N=Y
200 GOSUB 1000
210 PRINT "*";
```
Move o cursor mais Y espaços e imprime *.

```
220 LET N=R
230 GOSUB 1000
240 PRINT "I"
```
Move o cursor mais R espaços e imprime outro I. (Desta vez não há ponto e vírgula e o cursor muda de linha.)

```
●→
● 250 LET I$=INKEY$
```
Verifica se você está apertando alguma tecla.

```
260 IF I$<>"Q" THEN GOTO 290
```
Se a tecla não é Q, pula para 290.

```
270 LET Y=Y-1
280 LET R=R+1
```
Se a tecla é Q, Y aumenta de 1 e R diminui de 1; a estrela se move para a esquerda.

```
290 IF I$<>"P" THEN GOTO 320
```
Verifica se você apertou P.

```
300 LET Y=Y+1
310 LET R=R-1
```
Em caso afirmativo, a estrela se move para a direita.

```
320 IF Y<1 OR R<1 THEN GOTO 380
```
Verifica se você bateu em uma das paredes. Em caso afirmativo, pula para 380.

```
330 LET S=S+1
```
Aumenta sua posição de uma unidade.

```
340 IF S<M THEN GOTO 100
```
Volta para o início se sua posição é menor que M.

```
350 PRINT "MUITO BEM...VOCE CONSEGUIU"
360 PRINT "ATRAVESSAR O VALE DA MORTE!"
```
Chega aqui se você chegou à posição M sem bater.

```
370 STOP
380 PRINT "VOCE BATEU E SUA NAVE"
390 PRINT "SE DESINTEGROU"
400 STOP
1000 IF N=0 THEN RETURN
1010 FOR I=1 TO N
1020 PRINT " ";
1030 NEXT I
1040 RETURN
```
Sub-rotina para mover o cursor para os locais apropriados.

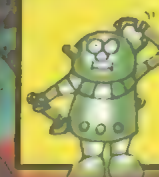
A listagem acima é para o TK-83. Para os outros computadores, faça as mudanças abaixo.

```
■ 100 LET D=INT(RND(0)*3-1)
● 100 LET D=INT(RND(1)*3-1)
▲■● 150
● 245 I$=""
● 250 IF PEEK(-16384)>127 THEN GET I$
```

Como tornar o jogo mais lento

Se o jogo é rápido demais para você, acrescente um "loop" de espera nas linhas 141 e 142:

```
141 FOR J=1 TO 100
142 NEXT J
```

Mude o número na linha 141 para ajustar a velocidade. Quanto menor o número, mais rápido o jogo.

Problema

Como você faria para tornar o vale mais comprido?

As Minas de Astron

Você acaba de ser eleito governador de uma colônia de mineração no Planeta Astron. Todas as decisões com relação à venda de minério, à compra de alimentos e à compra e venda de minas devem ser tomadas por você. Conseguirá manter a população satisfeita? Chegará ao fim do seu mandato de dez anos? Ou seu governo terminará em desastre?

Como o programa funciona

```
■● 10 LET L=INT(RND*3+5)
   20 LET P=INT(RND*60+40)
   30 LET M=INT(RND*50+10)*P
   40 LET FP=INT(RND*40+80)
   50 LET CE=INT(RND*40+80)
```
Escolhe o número de minas (L), o número de pessoas (P), a quantidade de dinheiro (X), o preço da comida (FP) e a quantidade de minério produzido por mina (CE) no início do jogo.

```
60 LET C=0
```
Faz a quantidade de minério armazenado igual a zero para começar.

```
70 LET S=1
```
Faz o fator de satisfação igual a 1.

```
80 LET Y=1
```
Faz o ano igual a 1.

```
■● 90 LET LP=INT(RND*2000+2000)
```
Escolhe o preço de compra e venda das minas.

```
■● 100 LET CP=INT(RND*12+7)
```
Escolhe o preço de venda do minério.

```
● 110 CLS
  120 PRINT "ANO";Y
  130 PRINT
  140 PRINT "EXISTEM ";P;" HABITANTES NA COLONIA"
  150 PRINT "VOCE DISPOE DE ";L;" MINAS E  $";M
  160 PRINT "O FATOR DE SATISFACAO E ";S
  170 PRINT
  180 PRINT "SUAS MINAS PRODUZIRAM ";CE;" TONELADAS CADA UMA"
  190 LET C=C+CE*L
  200 PRINT "MINERIO EM ESTOQUE:";C;" TONELADAS"
  210 PRINT "VENDAS"
  220 PRINT "PRECO DE VENDA DO MINERIO:$";CP;" POR TONELADA"
  230 PRINT "PRECO DE VENDA DAS MINAS :$";LP;" POR MINA"
```
Imprime a situação da colônia no momento.

```
240 PRINT "QUANTAS TONELADAS DE MINERIO
    VAI VENDER ?"
250 INPUT CS
260 IF CS<0 OR CS>C THEN GOTO 240
```
Pergunta quanto minério você quer vender e verifica se seu estoque é suficiente.

```
270 LET C=C-CS
```
Desconta a quantidade vendida da quantidade em estoque.

```
280 LET M=M+CS*CP
```
Calcula o valor do minério vendido e soma à sua reserva de dinheiro.

```
290 PRINT "QUANTAS MINAS VAI VENDER ?"
300 INPUT LS
310 IF LS<0 OR LS>L THEN GOTO 290
320 LET L=L-LS
330 LET M=M+LS*LP
```
Repete o processo para a venda de minas.

```
340 PRINT
350 PRINT "AGORA VOCE TEM $";M
360 PRINT
```
Imprime sua nova reserva de dinheiro.

```
370 PRINT "COMPRAS"
380 PRINT "QUANTO QUER GASTAR EM COMIDA ?
    (APROX. $100 /HAB.)"
390 INPUT FB
```
Pergunta quanto você quer gastar em comida e guarda em FB.

```
400 IF FB<0 OR FB>M THEN GOTO 380
```
Verifica se você tem dinheiro para pagar.

```
410 LET M=M-FB
```
Desconta da sua reserva de dinheiro.


```
420 IF FB/P>120 THEN LET S=S+.1
430 IF FB/P<80 THEN LET S=S-.2
```
Ajusta o fator de satisfação de acordo com o que você gastou em comida.

```
440 PRINT "QUANTAS MINAS QUER COMPRAR ?"
450 INPUT LB
460 IF LB<0 OR LB*LP>M THEN GOTO 440
```
Pergunta quantas minas você quer comprar e verifica se tem dinheiro suficiente.

```
470 LET L=L+LB
```
Aumenta o número de minas, se necessário.

```
480 LET M=M-LB*LP
```
Desconta da sua reserva de dinheiro.

```
490 IF S<.6 THEN GOTO 660
```
Verifica qual é o fator de satisfação. Se for muito baixo, pula para 660 e o jogo acaba.

```
■● 500 IF S>1.1 THEN LET CE=CE+INT(RND*20+1)
```
Se S é alto, aumenta a quantidade de minério produzido por mina.

```
■● 510 IF S<.9 THEN LET CE=CE-INT(RND*20+1)
```
Se S é baixo, diminui a quantidade produzida.

```
520 IF P/L<10 THEN GOTO 680
```
Se há menos de 10 colonos por mina, o jogo acaba.

```
■● 530 IF S>1.1 THEN LET P=P+INT(RND*10+1)
```
Se S é alto, o número de colonos aumenta.

```
■● 540 IF S<.9 THEN LET P=P-INT(RND*10+1)
```
Se S é baixo, o número de colonos diminui.

```
550 IF P<30 THEN GOTO 700
```
Se restam menos de 30 colonos, o jogo acaba.

```
■● 560 IF RND>.01 THEN GOTO 590
   570 PRINT "VAZAMENTO RADIOATIVO...MUITOS MORTOS"
   580 LET P=INT(P/2)
```
Introduz uma pequena probabilidade de metade da população morrer.

```
590 IF CE<150 THEN GOTO 620
600 PRINT "EXCESSO DE PRODUCAO...QUEDA DOS PRECOS"
610 LET CE=INT(CE/2)
```
Se a produção por mina é alta, o preço do minério cai à metade.

```
620 LET Y=Y+1
630 IF Y<11 THEN GOTO 90
```
O número do ano aumenta de 1; se for menor que 11, o computador volta para 90.

```
640 PRINT "VOCE CHEGOU AO FIM DO MANDATO"
```
Chega aqui se você consegue chegar ao décimo ano.

```
650 STOP
660 PRINT "OS COLONOS SE REVOLTARAM"
670 STOP
680 PRINT "EXCESSO DE TRABALHO"
690 STOP
700 PRINT "POPULACAO INSUFICIENTE"
710 STOP
```

A listagem acima é para TK-83. Para outros computadores, faça as mudanças abaixo.

```
● 110 HOME
● 10,20,30,40,50,90,100,500,510,530,540,560
  mude RND para RND(1)
■ 10,20,30,40,50,90,100,500,510,530,540,560
  mude RND para RND(0)
```

Problema

Como você faria para o computador lhe perguntar se quer jogar de novo e somar a quantia com que você terminou à quantia inicial para o jogo seguinte?

O significado de ■ ● está na pág. 2.

Salvamento no Espaço

Você precisa fazer uma viagem urgente a um planeta distante, do outro lado da Galáxia, para entregar remédios. A viagem é tão longa que você tem que fazê-la em estado de hibernação. Antes de partir, o computador lhe pergunta quanta energia vai destinar aos motores, ao sistema de controle ambiental e à blindagem da nave.

Quando você acorda, o computador lhe informa sobre o que aconteceu durante a viagem; se tudo correu bem, você está em órbita em torno do planeta. Agora precisa distribuir a energia que resta entre os retrofoguetes e a blindagem térmica, de modo a fazer uma boa aterrissagem.

Se conseguir cumprir a missão, será promovido a Almirante do Espaço. Boa sorte!

```
10 CLS
20 PRINT "SALVAMENTO NO ESPACO"
30 PRINT
40 PRINT "VOCE PRECISA DE INSTRUCOES ?"
50 INPUT I$
60 IF I$(1)="S" THEN GOSUB 1000
70 LET D=INT(RND*800+101)
80 LET E=INT(RND*400+401)
90 LET T=INT(D/SQR(E/5)+.5)
100 PRINT "O PLANETA ESTA A ";D;" UNIDADES DE DISTANCIA"
110 PRINT "VOCE TEM ";E;" UNIDADES DE ENERGIA"
120 PRINT "E DEVE CHEGAR EM ";T;" DIAS"
130 PRINT
140 PRINT "DISTRIBUICAO DE ENERGIA"
150 PRINT "PARA OS MOTORES ?"
160 INPUT P
170 PRINT "PARA O CONTROLE AMBIENTAL ?"
180 INPUT L
190 PRINT "PARA A BLINDAGEM ?"
200 INPUT S
210 IF P+L+S>E THEN GOTO 140
220 LET X=E-P-L-S
230 LET V=INT(SQR(P))
240 LET T1=INT(D/V)
250 CLS
260 PRINT "SUA VELOCIDADE E ";V
270 PRINT "VOCE DORMIU ";T1;" DIAS"
280 PRINT
290 FOR I=1 TO INT(RND*5+6)
300 IF RND>.5 THEN GOTO 430
310 GOTO 320+INT(RND*4)*30
320 PRINT "CHUVA DE ASTEROIDES...BLINDAGEM DANIFICADA"
330 LET S=S-20-INT(RND*40+1)
340 GOTO 430
350 PRINT "DEFEITO NO COMPUTADOR...PARADA PARA REPAROS"
360 LET D=D+INT(RND*20+1)
370 GOTO 430
380 PRINT "DEFEITO NO MOTOR...VELOCIDADE REDUZIDA"
390 LET V=V-.5
400 GOTO 430
410 PRINT "RADIACAO COSMICA...DEFEITO NO CONTROLE AMBIENTAL"
420 LET L=L-20-INT(RND*40+1)
430 FOR J=1 TO 50
440 NEXT J
450 NEXT I
460 LET T1=INT(D/V)
470 CLS
480 PRINT "CHEGOU EM ";T1;" DIAS"
490 IF S<0 THEN PRINT "BLINDAGEM DESTRUIDA"
500 IF S<0 THEN PRINT "SUA NAVE EXPLODIU"
510 IF L<=0 THEN PRINT "CONTROLE AMBIENTAL INOPERANTE"
520 IF L<=0 THEN PRINT "VOCE MORREU"
530 IF V<=0 THEN PRINT "MOTORES INOPERANTES"
540 IF T1>T THEN PRINT "VOCE LEVOU TEMPO DEMAIS"
550 IF S<0 OR L<=0 OR V<=0 OR T1>T THEN STOP
560 LET G=INT(RND*10+5)
570 LET G$="ALTA"
580 IF G<12 THEN LET G$="MEDIA"
590 IF G<8 THEN LET G$="BAIXA"
600 LET A=INT(RND*10+5)
610 LET A$="DENSA"
620 IF A<12 THEN LET A$="MEDIA"
630 IF A<8 THEN LET A$="RAREFEITA"
640 PRINT
650 PRINT "VOCE AGORA ESTA EM ORBITA"
660 PRINT "ENERGIA DISPONIVEL:";X
670 PRINT "A GRAVIDADE E ";G$
680 PRINT "A ATMOSFERA E ";A$
690 PRINT
700 PRINT "ENERGIA PARA OS RETROFOGUETES ?"
710 INPUT B
720 PRINT "ENERGIA PARA A BLINDAGEM TERMICA ?"
730 INPUT S
740 IF B+S>X THEN GOTO 700
750 CLS
760 IF B>=G*10 THEN GOTO 790
```

O programa continua na página seguinte.

```
770 PRINT "VOCE FEZ UMA CRATERA NOVA"
780 GOTO 860
790 IF S>=A*10 THEN GOTO 820
800 PRINT "VOCE E UMA LINDA ESTRELA CADENTE"
810 GOTO 860
820 PRINT "VOCE POUSOU LINDAMENTE...PARABENS"
830 IF X-S-B>25 THEN GOTO 860
840 PRINT "PENA QUE VOCE NAO TENHA ENERGIA"
850 PRINT "SUFICIENTE PARA ABRIR A ESCOTILHA"
860 STOP
1000 PRINT
1010 PRINT "VOCE VAI EMBARCAR EM UMA MISSAO"
1020 PRINT "PARA LEVAR SUPRIMENTOS MEDICOS"
1030 PRINT "PARA UM PLANETA DISTANTE."
1040 PRINT "PRIMEIRO VOCE PRECISA PREPARAR"
1050 PRINT "A NAVE PARA A VIAGEM DISTRIBUINDO"
1060 PRINT "A ENERGIA DISPONIVEL ENTRE OS"
1070 PRINT "MOTORES, A BLINDAGEM E O CONTROLE"
1080 PRINT "AMBIENTAL. VOCE ENTAO VIAJA EM "
1090 PRINT "ESTADO DE HIBERNACAO E QUANDO "
1100 PRINT "ACORDA RECEBE UM RELATORIO SOBRE"
1110 PRINT "O QUE ACONTECEU NO CAMINHO."
1120 PRINT "AGORA VOCE PRECISA POUSAR NO"
1130 PRINT "PLANETA......................"
1140 PRINT "APERTE QUALQUER TECLA"
●1150 IF INKEY$="" THEN GOTO 1150
●1160 CLS
1170 RETURN
```

A listagem acima é para o TK-83. Para outros computadores, faça as mudanças abaixo.

```
■todos os RND para RND(0)
●todos os RND para RND(1)
●10,250,470,750,1160 HOME
■●60 IF LEFT$(I$,1)="S" THEN GOSUB 1000
■●310 ON INT(RND*4+1) GOTO 320,350,380,410
■●430 FOR J=1 TO 500
●1150 GET I$
■1150 GET I$ : IF I$="" THEN GOTO 1150
```


Como aumentar o jogo

Este jogo é feito de duas partes distintas. Na primeira, você viaja no espaço; na segunda, tenta pousar no planeta. Se quiser, você pode acrescentar uma terceira parte na qual faz uma viagem cheia de percalços desde o local de pouso até a sede da Cruz Vermelha Intergaláctica.

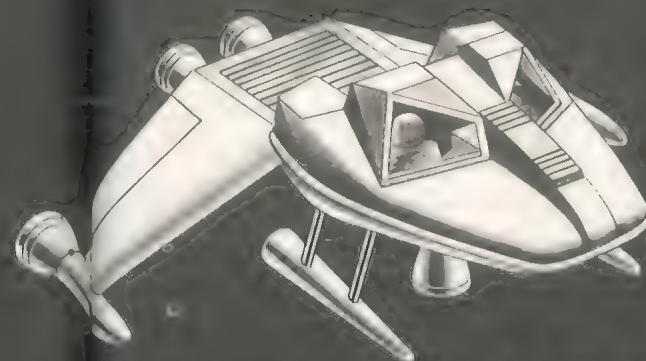
Pouso na Lua

Este jogo é diferente dos anteriores, pois utiliza instruções gráficas. Como essas instruções variam muito de micro para micro, apresentamos uma listagem separada para cada modelo. Leia a descrição do jogo nesta página e procure nas páginas seguintes a versão apropriada ao seu computador.

Como jogar Pouso na Lua

O Capitão Cometa, chefe dos pilotos espaciais, está sentado ao seu lado em uma pequena espaçonave de dois lugares que se aproxima rapidamente da superfície da Lua. Está na hora do teste final do Curso de Pilotagem Avançada. Você tem que pousar na Lua com a menor velocidade possível. Você pode controlar o empuxo do retrofoguete, apertando a tecla A para aumentá-lo e D para diminuí-lo, observando ao mesmo tempo o pouso na tela do computador. Se aumentar demais o empuxo, a nave começará a subir. Se diminuir demais, acabará abrindo uma nova cratera na Lua. Que nota lhe dará o Capitão Cometa?

Pouso na Lua: Versão TRS-80

```
10 CLS
20 CLEAR 200
30 B$=STRING$(25,131)
40 M1$=CHR$(194)+STRING$(2,176)
50 M2$=" "+STRING$(4,191)
60 M3$=CHR$(131)+CHR$(135)+STRING$
   (2,131)+CHR$(139)+CHR$(131)
70 GOSUB 260
80 GOSUB 300
90 C=1:GOSUB 390
100 A=1:B=F:GOSUB 440
110 A=2:B=ABS(V):GOSUB 440
120 A=3:B=H:GOSUB 440
130 A=4:B=T:GOSUB 440
140 GOSUB 480
150 V1=V-T/20+G:F=F-T/10
160 H1=H-(V+V1)/10
170 C=0:GOSUB 390
180 IF H1<0 THEN 230
190 H=H1:V=V1
200 IF H<=100 THEN 90
210 GOSUB 530
220 GOTO 250
230 H=0:C=1:GOSUB 390
240 GOSUB 590
250 END
260 H=100:F=100:T=0
270 V=INT(RND(0)*10+6)
280 G=INT(RND(0)*40+41)/100
290 RETURN
300 FOR X=80 TO 127
310 SET (X,47-INT(RND(0)*5))
320 NEXT
330 PRINT "GRAVIDADE:";G
340 PRINT @192,"COMBUSTIVEL:"
350 PRINT @384,"VELOCIDADE:"
360 PRINT @576,"ALTITUDE:"
370 PRINT @768,"EMPUXO:"
380 RETURN
390 Y=818-64*INT(H/8)
400 PRINT @Y,;:IF C=1 THEN PRINT
    M1$;ELSE PRINT CHR$(196);
410 PRINT @Y+64,;:IF C=1 THEN
    PRINT M2$;ELSE PRINT CHR$(198);
420 PRINT @Y+128,;:IF C=1 THEN
    PRINT M3$;ELSE PRINT CHR$(198)
430 RETURN
440 Y=(A*3+1)*64
450 PRINT @Y,CHR$(217);
460 PRINT @Y,LEFT$(B$,B/4);
470 RETURN
480 I$=INKEY$
490 IF I$="A" THEN T=T+4:IF
    T>100 THEN T=100
500 IF I$="D" THEN T=T-4:IF
    T<0 THEN T=0
510 IF T>F THEN T=F
520 RETURN
530 CLS
540 FOR I=1 TO 20
550 PRINT @INT(RND(0)*1024),"*"
560 NEXT
570 PRINT @470,"PERDIDOS NO ESPACO"
580 RETURN
590 PRINT "POUSOU COM VELOCIDADE ";
    INT((V+V1)*5)/10
600 IF (V+V1)<8 THEN PRINT "A SALVO"
    ELSE PRINT "TODOS MORTOS"
610 RETURN
```


Pouso na Lua: Versão TK-85

```
20 DEF FNR(X)=INT(RND*X+1)
30 GOSUB 260
40 GOSUB 300
50 LET C=0:GOSUB 380
60 LET A=1:LET B=F:LET C=2*(F<25)
70 GOSUB 440
80 LET A=2:LET B=ABS V:
   LET C=4*(V<0)
90 GOSUB 440
100 LET A=3:LET B=H:LET C=2*(H<25)
110 GOSUB 440
120 LET A=4:LET B=T:LET C=0
130 GOSUB 440
140 GOSUB 500
150 LET V1=V-T/20+G:LET F=F-T/10
160 LET H1=H-(V+V1)/10
170 LET C=1:GOSUB 380
180 IF H1<0 THEN GOTO 230
190 LET H=H1:LET V=V1
200 IF H<=100 THEN GOTO 50
210 GOSUB 550
220 GOTO 250
230 LET H=0:LET C=0:GOSUB 380
240 GOSUB 610
250 STOP
260 LET H=100:LET F=100:LET T=0
270 LET V=5+FNR(10)
280 LET G=(FNR(40)+40)/100
290 RETURN
300 PLOT 180,8
310 FOR X=1 TO 15
320 DRAW 5,FNR(3)-2
330 NEXT X
340 PRINT "GRAVIDADE:";G
350 PRINT ''"COMBUSTIVEL:"'''"VEL:"
360 PRINT '''"ALTITUDE:"''''"EMPUXO:"
370 RETURN
380 INVERSE C
390 LET Y=H*1.3+10
400 PLOT 200,Y:DRAW 34,0
410 DRAW -4,20:DRAW -13,10
420 DRAW -13,-10:DRAW -4,-20
430 RETURN
440 LET Y=172-A*32
450 INK C
460 PL[illegible],Y
470 DRA[illegible]B,0
480 DRAW INVERSE 1,100-B,0
490 RETURN
500 LET I$=INKEY$
510 IF I$="A" THEN LET T=T+4:
    IF T>100 THEN LET T=100
520 IF I$="D" THEN LET T=T-4:
    IF T<0 THEN LET T=0
530 IF T>F THEN LET T=F
540 RETURN
550 CLS
560 FOR I=1 TO 20
570 PRINT AT FNR(21),FNR(31);"*"
580 NEXT I
590 PRINT "PERDIDOS NO ESPACO!!!"
600 RETURN
610 PRINT AT 0,0;"POUSOU COM ";
    INT((V+V1)*5)/10
620 IF (V+V1)<8 THEN GOTO 640
630 PRINT "TODOS MORTOS":RETURN
640 PRINT "A SALVO":RETURN
```

Pouso na Lua: Versão TK-83

```
10 CLS
20 LET B$="[66666666666666666666]"
30 GOSUB 360
40 GOSUB 420
50 LET C=1
60 GOSUB 490
70 LET A=1
80 LET B=F
90 GOSUB 590
100 LET A=2
110 LET B=ABS(V
120 GOSUB 590
130 LET A=3
140 LET B=H
150 GOSUB 590
160 LET A=4
170 LET B=T
180 GOSUB 590
190 GOSUB 630
200 LET V1=V-T/20+G
210 LET F=F-T/10
220 LET H1=H-(V+V1)/10
230 LET C=0
240 GOSUB 490
250 IF H1<0 THEN GOTO 310
260 LET H=H1
270 LET V=V1
280 IF H<=100 THEN GOTO 50
290 GOSUB 700
300 GOTO 350
310 LET H=0
320 LET C=1
330 GOSUB 490
340 GOSUB 760
350 STOP
360 LET H=100
370 LET F=100
380 LET T=0
390 LET V=5+INT(RND*10+1)
400 LET G=INT(RND*40+40)/100
410 RETURN
420 PRINT AT 21,20;"[SDADSADAS]"
430 PRINT AT 0,0;"GRAVIDADE:";G
440 PRINT AT 3,0;"COMBUSTIVEL:"
450 PRINT AT 7,0;"VELOCIDADE:"
460 PRINT AT 11,0;"ALTITUDE:"
470 PRINT AT 15,0;"EMPUXO:"
480 RETURN
490 LET Y=(100-H)/6+1
500 IF C=0 THEN GOTO 550
510 PRINT AT Y,24;"[34]"
520 PRINT AT Y+1,23;"[Q  W]"
530 PRINT AT Y+2,23;"[TGGY]"
540 GOTO 580
550 PRINT AT Y,24;"  "
560 PRINT AT Y+1,23;"    "
570 PRINT AT Y+2,23;"    "
580 RETURN
590 LET Y=A*4          (20 ESPAÇOS)
600 PRINT AT Y,0;"                    "
610 PRINT AT Y,0;B$( TO B/5)
620 RETURN
630 LET I$=INKEY$
640 IF I$="A" THEN LET T=T+4
650 IF T>100 THEN LET T=100
660 IF I$="D" THEN LET T=T-4
670 IF T<0 THEN LET T=0
680 IF T>F THEN LET T=F
690 RETURN
700 CLS
710 FOR I=1 TO 20
720 PRINT AT INT(RND*22),
    INT(RND*32);"*"
730 NEXT I
740 PRINT AT 10,10;"PERDIDOS
    NO ESPACO"
750 RETURN
760 PRINT AT 20,0;"POUSOU
    COM VEL ";INT((V+V1)*5)/10
770 IF (V+V1)<8 THEN PRINT
    "A SALVO"
780 IF (V+V1)>=8 THEN PRINT
    "TODOS MORTOS"
790 RETURN
```

Nota especial para os usuários do TK-83

Os caracteres graficos sao indicados na listagem de acordo com a seguinte convencao: um carater '[' significa aperte a tecla GRAPHICS e depois escreva as letras ou numeros seguintes com a tecla SHIFT apertada ate chegar a um ']'. Entao aperte de novo a tecla GRAPHICS, voltando ao modo normal.

Pouso na Lua: Versão Apple

```
10 HOME
20 HGR
30 DEF FNR(X)=INT(RND(1)*X+1)
40 GOSUB 230
50 GOSUB 270
60 C=3:GOSUB 380
70 A=1:B=F:GOSUB 460
80 A=2:B=AABS(V):GOSUB 460
90 A=3:B=H:GOSUB 460
100 A=4:B=T:GOSUB 460
110 GOSUB 520
120 V1=V-T/20+G:F=F-T/10
130 H1=H-(V+V1)/10
140 C=0:GOSUB 380
150 IF H1<0 THEN 200
160 H=H1:V=V1
170 IF H<=100 THEN 60
180 GOSUB 570
190 GOTO 220
200 H=0:C=3:GOSUB 380
210 GOSUB 600
220 END
230 H=100:F=100:T=0
240 V=5+FNR(10)
250 G=(FNR(40)+40)/100
260 RETURN
270 HCOLOR=3
280 HPLOT 0,155
290 FOR X=0 TO 279 STEP 5
300 HPLOT TO X,159-FNR(10)
310 NEXT
320 FOR I=1 TO 30:HPLOT FNR(279),
    FNR(150)
330 NEXT
340 VTAB 21:PRINT TAB(34);"G=";G
350 VTAB(21):PRINT "COMBUSTIVEL:":
    PRINT "VEL:"
360 PRINT "ALTITUDE:":PRINT "EMPUXO:";
370 RETURN
380 HCOLOR=C
390 Y=(100-H)*1.3
400 HPLOT 140,Y TO 120,Y+10
410 HPLOT TO 120,Y+20:
    HPLOT TO 160,Y+20
420 HPLOT TO 160,Y+10:HPLOT TO 140,Y
430 HPLOT 155,Y+20 TO 160,Y+25
440 HPLOT 125,Y+20 TO 120,Y+25
450 RETURN
460 VTAB (20+A):HTAB 8
470 INVERSE
480 PRINT SPC(B/4);
490 NORMAL
500 PRINT SPC(26-B/4);
510 RETURN
520 I$="":IF PEEK(-16384)>127
    THEN GET I$
530 IF I$="A" THEN T=T+4:
    IF T>100 THEN T=100
540 IF I$="D" THEN T=T-4:
    IF T<0 THEN T=0
550 IF T>F THEN T=F
560 RETURN
570 HOME:VTAB 21
580 PRINT "PERDIDOS NO ESPACO!!"
590 RETURN
600 HOME:VTAB 22
610 PRINT "POUSOU COM VEL ":
    INT((V+V1)*5)/10
620 IF (V+V1)<8 THEN 640
630 PRINT "MORRERAM TODOS":RETURN
640 PRINT "A SALVO":RETURN
```

Algumas sugestões

Aqui estão algumas idéias que você poderá usar nos programas deste livro ou em seus próprios programas.

Ao incluir novas instruções em um programa já pronto, você tem duas opções: utilizar apenas linhas vagas ou numerar de novo todo o programa. No segundo caso, não se esqueça de mudar todas as instruções GOTO e GOSUB.

Faça o computador explicar como se joga

Você pode fazer com que o computador imprima instruções sobre o jogo. Para isso, basta acrescentar algumas linhas no início do programa, como no exemplo abaixo, e colocar uma sub-rotina no final.

```
   10 PRINT "NOME DO JOGO"
   11 PRINT "VOCE QUER SABER"
   12 PRINT "COMO JOGAR ?"
   15 INPUT I$
TK 17 IF I$(1)="S" THEN GOSUB 1000
■● 17 IF LEFT$(I$,1)="S" THEN GOSUB 1000
```

o programa principal entra aqui

```
1000 PRINT "VOCE TEM QUE FAZER"
1010 PRINT "O SEGUINTE : ....."
1999 RETURN
```

Você pode usar quantas linhas quiser para as instruções. Não se esqueça de colocar um número e a palavra PRINT no início de cada linha. O número de caracteres entre aspas deve ser menor ou igual ao número de caracteres que o seu computador pode imprimir em uma linha. Não se esqueça também de colocar uma linha RETURN no final da sub-rotina.

Faça o computador parar e esperar por você

Se houver muitas linhas de instruções, talvez você precise incluir a sub-rotina abaixo, que faz com que o computador pare de executar o programa até você apertar uma tecla. Assim, evitará que as instruções desapareçam na parte superior da tela antes que tenha tempo de lê-las. Coloque uma linha GOSUB no ponto em que deseja que o programa pare e acrescente no final a seguinte sub-rotina:

```
     1000 PRINT "APERTE UMA TECLA"
     1010 PRINT "PARA CONTINUAR"
■ TK 1020 IF INKEY$="" THEN GOTO 1020
●    1020 GET I$
     1030 PRINT
     1040 RETURN
```

Faça o computador "conversar" com você

Você pode programar o computador para fazer perguntas e agir de acordo com as suas respostas. Aqui está, por exemplo, um pequeno programa que faz com que o computador se recuse a jogar a menos que seu nome comece com a letra J:

```
   1 PRINT "COMO VOCE SE CHAMA?"
   2 INPUT I$
TK 3 IF I$(1)<>J" THEN GOTO 1000
■● 3 IF LEFT$(I$,1)<>"J" THEN GOTO 1000
   4 PRINT "OK...VOCE PODE JOGAR"
   5 PRINT "ESTA PRONTO ?"
   6 INPUT J$
TK 7 IF J$(1)<>"S" THEN GOTO 5
■● 7 IF LEFT$(J$,1)<>"S" THEN GOTO 5
```

O programa principal entra aqui

```
1000 PRINT "DESCULPE.ESTE JOGO E"
1010 PRINT "SO PARA PESSOAS"
1020 PRINT "CUJOS NOMES COMECAM"
1030 PRINT "COM A LETRA J"
```


Aqui está um programa em que o computador desconfia de que você não tem coragem suficiente para jogar.

```
   10 PRINT "JOGO MUITO PERIGOSO"
   12 PRINT "VOCE TEM CORAGEM"
   14 PRINT "DE ENFRENTAR"
   15 PRINT "O MONSTRO CABELUDO?"
   16 INPUT I$
TK 17 IF I$(1)="S" THEN GOTO 20
■● 17 IF LEFT$(I$,1)="S" THEN GOTO 20
   18 PRINT "COVARDE!!!"
   19 STOP
```

Você pode combinar o programa acima com a sub-rotina de instruções colocando as linhas 11 a 17 do programa da página anterior como linhas 20 a 26 do programa acima. Comece o programa principal na linha 30 e acrescente a sub-rotina de instruções no final.

Quer jogar de novo?

Ao invés de escrever RUN cada vez que deseja começar um jogo, você pode fazer com que o computador pergunte se quer jogar de novo. Coloque as linhas abaixo no final do programa, antes do último STOP.

```
   1000 PRINT "QUER JOGAR DE NOVO ?"
   1010 INPUT I$
TK 1020 IF I$(1)="S" THEN RUN
■● 1020 IF LEFT$(I$,1)="S" THEN RUN
   1030 PRINT "OI...TCHAU"
```

Mude os números das linhas de acordo com o seu programa.

Efeitos sonoros

Muitos microcomputadores dispõem de recursos para gerar sons. Você pode utilizar esses recursos em qualquer ponto dos seus programas. Experimente, por exemplo, simular o ruído de uma explosão ou fazer com que o computador toque uma musiquinha toda vez que alguém ganhar. Como as instruções para gerar sons variam de computador para computador, você terá que consultar o manual do seu micro. Em certos casos, você pode acrescentar uma única linha ao programa no local onde deseja o som. Em outros, vai precisar de várias linhas; nesse caso, é melhor usar uma sub-rotina.

A linha abaixo mostra, por exemplo, a instrução para produzir o som do dó central do piano durante 6 segundos em um computador do tipo TRS-80:

```
SOUND 89,100
```

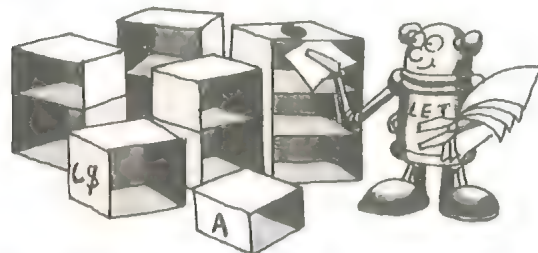
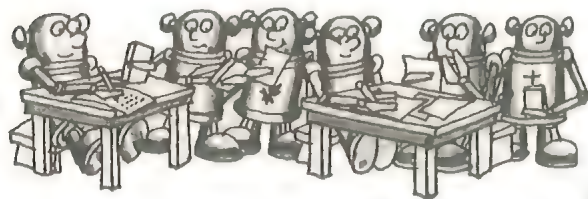
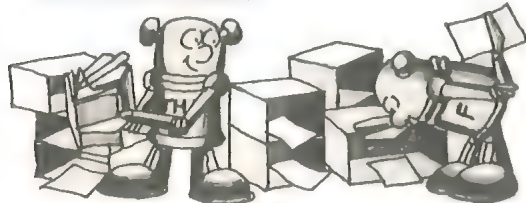
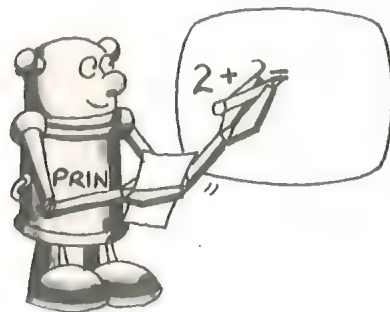
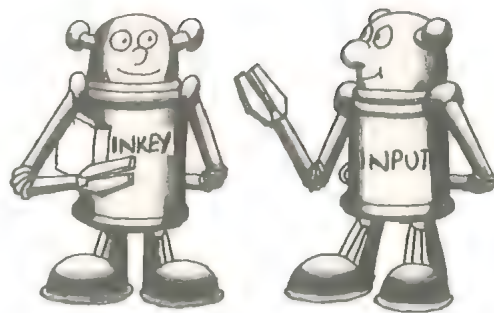
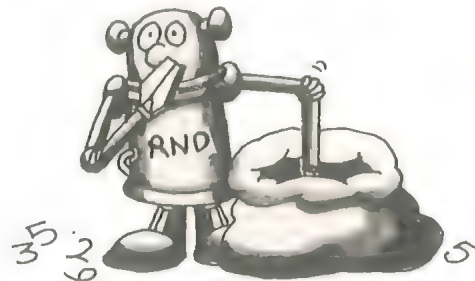
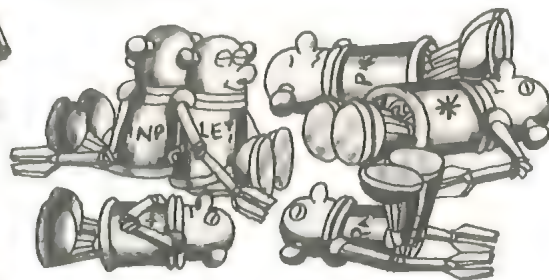
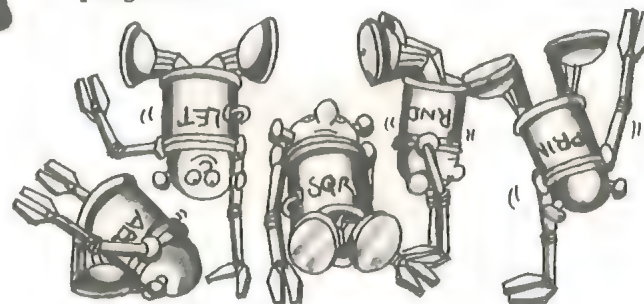
Você pode usar quantas instruções quiser para gerar sons, mas não se esqueça de colocar um número no início de cada instrução.

Nota especial para os usuários do TK-85

Se o seu micro é um TK-85, talvez alguns dos jogos deste livro sejam rápidos demais para o seu computador. As instruções para diminuir a velocidade aparecem na mesma página que o jogo correspondente. Como regra geral, quanto maior o número que você usar no "loop" de espera, mais lento será o jogo.

Como programar jogos

Depois que você estiver familiarizado com os jogos deste livro, certamente terá vontade de programar outros jogos. Nestas duas páginas, discutimos alguns conceitos básicos de que vai necessitar para isso.

Antes de mais nada, vale a pena pensar um pouco no que o seu computador pode fazer e no que ele não pode.

* Pode guardar informações.

* Pode fazer contas.

* Pode tomar decisões comparando de várias formas informações anteriores.

* Pode mostrar os resultados dos cálculos e decisões e também o que está guardado na memória.

* Pode pedir novas informações.

* Pode escolher um número ao acaso, usando RND.

* Não pode fazer nada que não esteja no programa.

* Se você usar instruções válidas de BASIC, não pode distinguir instruções "sensatas" de instruções tolas.

Quando estiver planejando o programa de um jogo, não inclua nada que o seu micro não seja capaz de fazer.

Como planejar um jogo

Antes de programar o computador para jogar um jogo, você precisa saber exatamente como será jogado e quais serão as regras. Como o computador vai necessitar de uma série de instruções simples e lógicas, é melhor imaginar primeiro todo o jogo e depois dividi-lo em pequenas partes.

Em seguida, você deve escrever todas as partes do jogo, na ordem em que serão executadas (escreva em português; ainda não está na hora de usar BASIC).

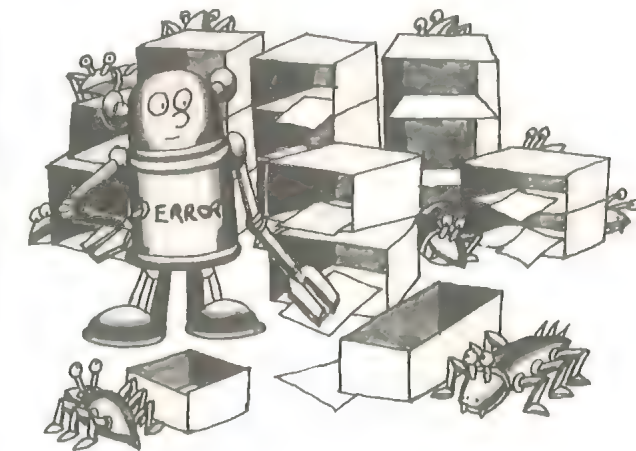
A ilustração abaixo mostra o plano de um jogo simples de atirar, que pode ser usado para disparar balas de canhão contra um navio pirata ou raios *laser* contra uma nave alienígena.

PLANO DE JOGO

1) IMPRIMIR TÍTULO E INSTRUÇÕES
2) ESCOLHER UM ALVO
3) COMEÇAR UM "LOOP" QUE PERMITA AO JOGADOR N TENTATIVAS
4) PEDIR AO JOGADOR PARA DAR UM TIRO
5) VERIFICAR SE O TIRO ACERTOU
6) IMPRIMIR UMA MENSAGEM SOBRE O RESULTADO DO TIRO
7) VOLTAR PARA A TENTATIVA SEGUINTE

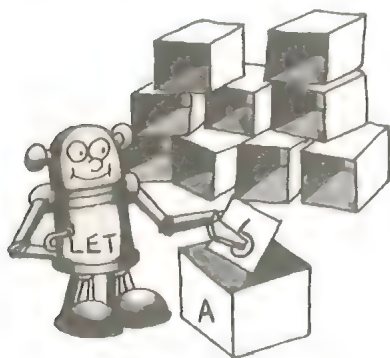
Como escrever o programa

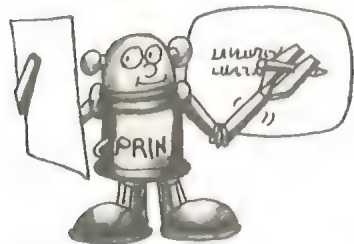
O passo seguinte consiste em traduzir o seu plano para a linguagem BASIC. Cada parte do plano pode corresponder a várias linhas de BASIC. Deixe números vagos entre os números das linhas do programa, para poder acrescentar mais tarde novas instruções, se for necessário.

Escreva um rascunho do programa e comece a testá-lo no computador. O computador é capaz de detectar muitos erros de programação e mesmo, em certos casos, de indicar exatamente onde está o erro. Se o seu programa não der certo logo da primeira vez, não desanime; "acertar" um programa é um processo lento e tedioso, até mesmo para os programadores experientes.

Depois que o programa básico estiver funcionando, você pode aperfeiçoá-lo, acrescentando mensagens, comentários, alvos diferentes etc. Também pode incluir partes dos programas deste livro nos seus jogos.

Não espere que os seus primeiros jogos sejam muito movimentados e originais. Comece com idéias simples e esteja preparado para modificá-las se for necessário. É provável que você descubra que incluiu alguma coisa no jogo que é muito fácil para uma pessoa mas muito difícil para um computador. Quando adquirir experiência, saberá instintivamente o que o seu computador é capaz de fazer.

Instruções de BASIC

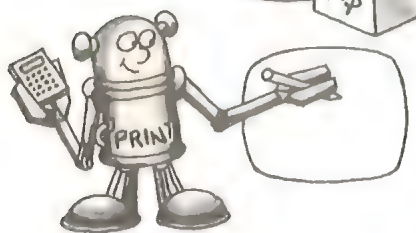
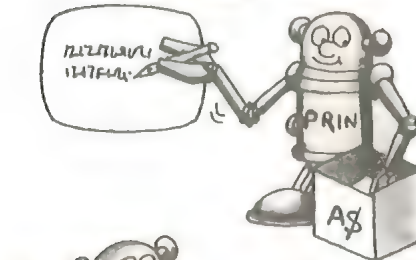
Este capítulo explica como funcionam algumas instruções comuns de BASIC. Quase todas foram usadas nos programas deste livro, de modo que você pode consultar as listagens dos jogos para ver o que essas instruções fazem nos programas. Nem todas as instruções podem ser usadas em todos os micros mencionados neste livro. A tabela de equivalência da página 44 mostra as diferenças entre os vários modelos.

LET diz ao computador para dar um nome a uma parte da memória e guardar nesta parte um número que você especifica. Assim, por exemplo, LET A=6 significa que o computador deve chamar de "A" uma parte da memória e guardar nesta parte o número 6. "A" é chamado de "variável" e colocar alguma coisa em A é chamado de "atribuir um valor a uma variável".

Alguns nomes de variáveis são seguidos por um cifrão (A$, por exemplo). Isto significa que a variável é do tipo "string" e a memória correspondente pode conter um número qualquer de caracteres (letras, números e símbolos).

PRINT diz ao computador para mostrar alguma coisa na tela. Esta instrução pode ser usada de várias formas:

Uma mensagem entre aspas precedida por PRINT é mostrada na tela exatamente como foi escrita. A parte entre aspas não precisa ser em BASIC, pode ser qualquer coisa.

A instrução PRINT seguida pelo nome de uma variável (PRINT A ou PRINT A$, por exemplo) diz ao computador para mostrar na tela o valor da variável.

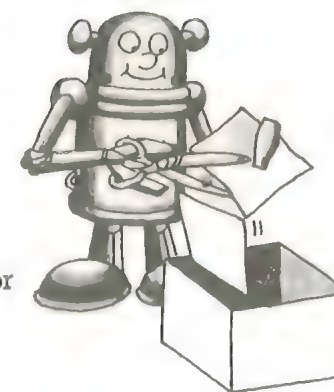
PRINT também pode ser usada para fazer cálculos e mostrar o resultado. Assim, por exemplo, a instrução PRINT 6*4 faz o computador mostrar na tela o número 24.

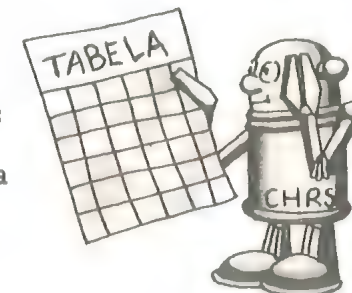
Você pode usar PRINT sozinha para deixar uma linha em branco na tela.

RND diz ao computador para escolher um número ao acaso. Computadores diferentes usam diferentes formas de RND (essas diferentes formas estão na tabela da pág. 44). Nos computadores TK, RND gera um número entre o 0 e 0,99999999. Você pode mudar os limites dos números gerados por RND multiplicando-os por uma constante e somando-os a outra constante. Assim, por exemplo, RND*20 gera um número entre 0 e 19,99999999, enquanto que RND*20+1 gera um número entre 1 e 20,99999999.

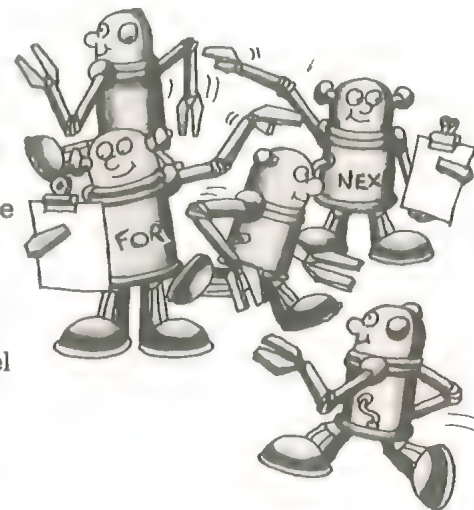
Para gerar apenas números inteiros, use a instrução INT.

Para gerar letras e símbolos, use a instrução CHR$.

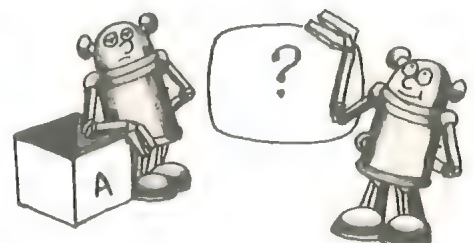
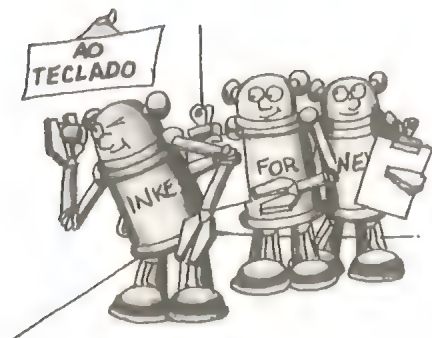
INT é a abreviação de inteiro. No caso de números positivos, diz ao computador para ignorar tudo que está à direita do ponto decimal. Assim, por exemplo, INT (20.999) é o número 20. No caso de números negativos, tudo que está à direita do ponto é ignorado e o número que está à esquerda é "aumentado" de uma unidade. Assim, por exemplo, INT(—3.6) é o número —4.

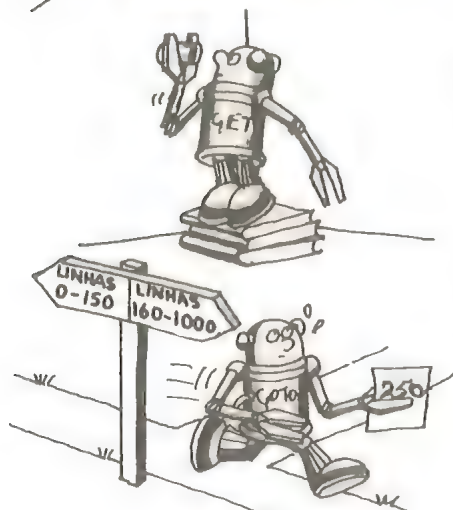
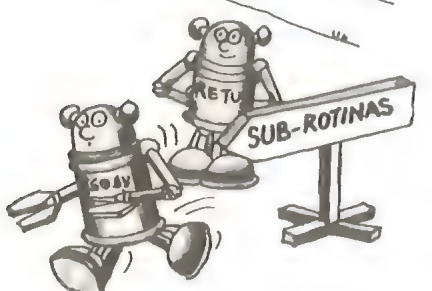
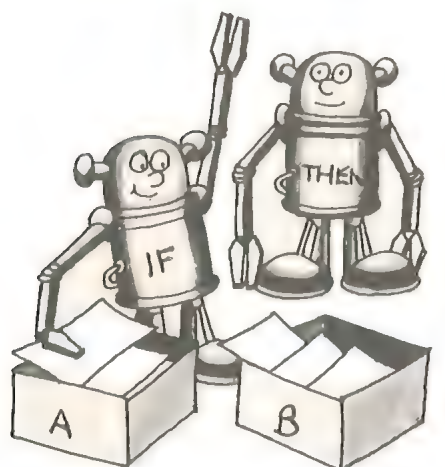
A instrução INT é freqüentemente usada junto com RND, como em INT(RND*20 + 1), que diz ao computador para escolher ao acaso um número inteiro entre 1 e 20.

CHR$ é usada para transformar números em letras. Com exceção do TK-82C e TK-83, todos os computadores mencionados neste livro utilizam o código ASCII* para a correspondência entre números e caracteres. Assim, por exemplo, o número de código para a letra A é 65 e a instrução CHR$(65) faz a letra A aparecer na tela.

Você pode usar as instruções CHR$, INT e RND para fazer o computador escolher letras ao acaso, como em CHR$(INT(RND*26+65)), que no computador TK-85 gera uma letra qualquer entre A e Z (para os outros computadores, consulte a tabela da página 44).

FOR é usada para começar um "loop" que faz o computador repetir uma parte do programa um certo número de vezes. Deve ser seguida pelo nome de uma variável (G, por exemplo) e pelos valores inicial e final da mesma variável (=1 TO 10, por exemplo).

O final do "loop" é indicado pela instrução NEXT (NEXT G, neste exemplo), que faz o valor da variável aumentar de uma unidade e manda o computador de volta para a instrução FOR. Quando o valor da variável chega ao valor final, o computador ignora a instrução NEXT e passa para a linha seguinte. Para cada instrução FOR deve haver uma instrução NEXT, caso contrário o programa não funcionará.

INPUT diz ao computador para dar um nome a uma variável, mostrar na tela um ponto de interrogação e esperar que você escreva o valor da variável. A execução do programa só prossegue se você apertar a tecla RETURN, ENTER ou NEWLINE.

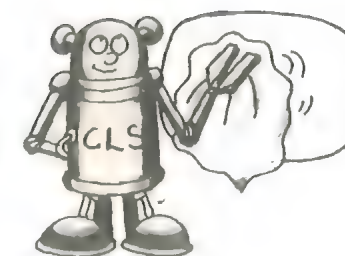
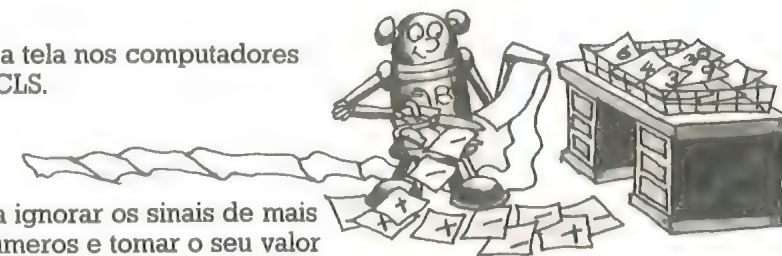
Você pode usar variáveis numéricas (sem o cifrão) ou variáveis tipo "string" (com um cifrão no final) em uma instrução INPUT, mas se usar uma variável numérica o computador não aceitará letras.

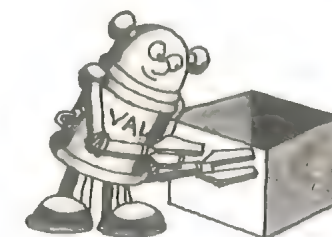
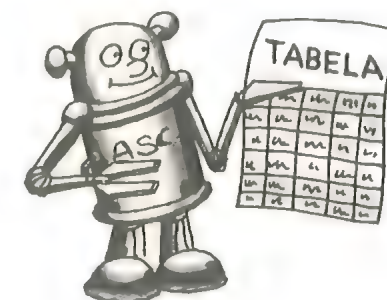
*O código completo está na pág. 43.

**INKEY$** diz ao computador para verificar se você está apertando alguma tecla e, em caso afirmativo, qual é essa tecla. O computador não espera você apertar alguma tecla, como no caso da instrução INPUT. A instrução INKEY$ é geralmente usada dentro de um "loop" que faz o computador verificar muitas vezes o teclado. Isto é necessário porque os computadores funcionam muito depressa e você não teria tempo de apertar uma tecla no tempo que o computador leva para fazer uma única verificação.

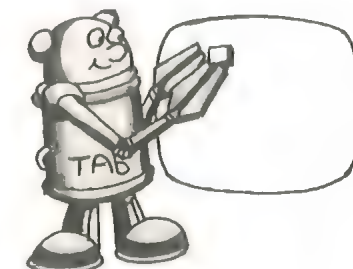
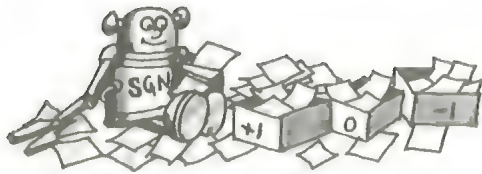
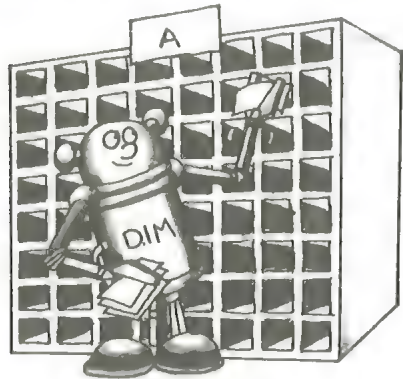
Quando o "loop" termina sem que você tenha apertado nenhuma tecla, o computador prossegue com um "string" que não contém nenhum símbolo (o chamado "string" vazio).

GOTO faz o computador pular para uma instrução mais acima ou mais abaixo, ignorando as instruções intermediárias. GOTO deve sempre ser seguida pelo número da linha para onde o computador deve pular.

GOSUB diz ao computador para sair do programa principal e ir para uma sub-rotina. GOSUB deve ser seguida pelo número da primeira linha da sub-rotina. A última instrução de qualquer sub-rotina deve ser RETURN, que manda o computador de volta ao programa principal, para a linha seguinte à da instrução GOSUB.

IF...THEN diz ao computador para verificar se uma expressão é verdadeira ou falsa e fazer coisas diferentes, dependendo da resposta. É usada com os seguintes sinais, além das palavras AND e OR:

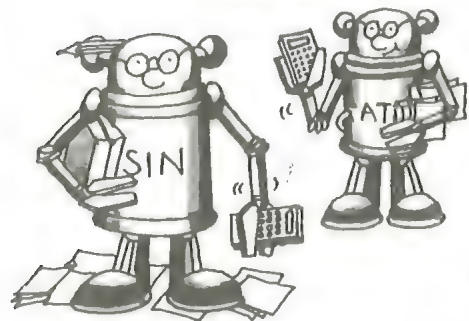
=	igual a
<	menor que
>	maior que
< =	menor ou igual a
> =	maior ou igual a
< >	diferente de

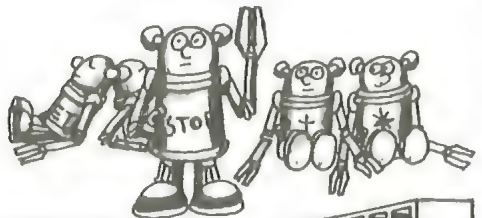
Se o computador verifica que a expressão é verdadeira, executa a instrução que se segue à palavra THEN. Se a expressão é falsa, ignora o resto da linha e passa para a linha seguinte.

CLS é usada para apagar tudo que está na tela, sem afetar o que se encontra na memória do computador. É útil para limpar a tela antes de executar um programa ou para tentar uma mensagem na tela apenas por um período limitado de tempo durante um jogo.

HOME é usada para limpar a tela nos computadores do tipo Apple, em lugar de CLS.

ABS diz ao computador para ignorar os sinais de mais e de menos na frente dos números e tomar o seu valor "absoluto". Assim, por exemplo, ABS(−10) é 10 e ABS(+10) também é 10.

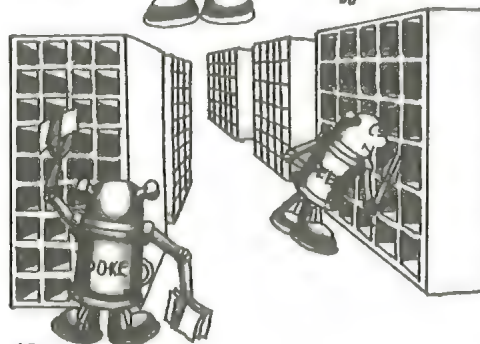
VAL é usada para obter o valor numérico de um número associado a uma variável "string". Em outras palavras, a instrução VAL diz ao computador para ignorar o cifrão e considerar a variável como se fosse uma variável numérica. Assim, por exemplo, se I$="60", então VAL(I$) é o número 60.

ASC é usada para obter o número do código ASCII correspondente a um dado número, letra ou símbolo. Assim, por exemplo, ASC ("3") é igual a 51. A expressão entre parênteses deve ser do tipo "string", como "20" ou A$.
Os micros TK não têm a instrução ASC, embora o TK-85 use o código ASCII.

CODE é a instrução usada nos micros TK em lugar de ASC. Como no caso da instrução ASC, CODE deve ser sempre seguida por uma expressão do tipo "string", entre parênteses. Observe que os números de código dos caracteres nos micros TK-82 e TK-83 não são os mesmos que nos outros computadores.

TAB é uma instrução usada para mover o cursor na tela até a coluna desejada. Geralmente é usada em conjunto com PRINT para mostrar alguma coisa no meio da tela. O número de espaços de deslocamento do cursor é colocado entre parênteses depois de TAB. O número máximo que pode ser usado depende do modelo de computador (vide tabela na pág. 45).

SGN diz ao computador para verificar o sinal de um número. O resultado é −1 se o número é negativo, 0 se o número é zero e +1 se o número é positivo. Assim, por exemplo, SGN (−30) é −1, SGN(7) é +1 e SGN (0) é 0.

DIM diz ao computador para reservar espaço na memória para um "array" (variável indexada). Assim, por exemplo, DIM X(6) diz ao computador para reservar um espaço suficientemente grande para conter 6 elementos, que serão chamados de X(1), X(2), X(3), etc. DIM A(8,8) reserva um espaço na memória para 64 elementos, que serão chamados de A(1,1), A(1,2), etc. Uma variável indexada nunca pode ser usada em um programa com um número entre parênteses maior que o especificado na instrução DIM.

SQR é usada para obter a raiz quadrada de um número. Assim, por exemplo, SQR(16) é igual a 4.

SIN é usada para calcular o seno de um ângulo. Em um triângulo retângulo, o comprimento do lado oposto a um ângulo dividido pelo comprimento da hipotenusa (lado oposto ao ângulo reto) é chamado de seno desse ângulo. Para usar SIN em um programa, o ângulo deve ser expresso em radianos, e não em graus.

ATN é uma das funções trigonométricas que os computadores são capazes de calcular (outra é o seno, de que falamos acima). ATN é a abreviação de arco tangente. Em alguns computadores, esta instrução é chamada de ARCTAN.

STOP diz ao computador para parar de executar o programa. Em todos os computadores, a não ser o TK-82C e o TK-83, a instrução END faz o mesmo efeito.

PEEK é usada para descobrir o que existe em uma certa região da memória do computador. Esta instrução é sempre seguida por um número entre parênteses, que especifica um "endereço" na memória.

POKE é uma forma especial de guardar informações na memória do computador, usando o "endereço" de uma memória.

Código ASCII

Número de código	Caractere correspondente	Número de código	Caractere correspondente
32	espaço	62	>
33	!	63	?
34	"	64	@
35	#	65	A
36	$	66	B
37	%	67	C
38	&	68	D
39	'	69	E
40	(	70	F
41	)	71	G
42	*	72	H
43	+	73	I
44	,	74	J
45	−	75	K
46	.	76	L
47	/	77	M
48	0	78	N
49	1	79	O
50	2	80	P
51	3	81	Q
52	4	82	R
53	5	83	S
54	6	84	T
55	7	85	U
56	8	86	V
57	9	87	W
58	:	88	X
59	;	89	Y
60	<	90	Z
61	=		

Código TK-82C/TK-83

Número de código	Caractere correspondente	Número de código	Caractere correspondente
11	"	41	D
12	£	42	E
13	$	43	F
14	:	44	G
15	?	45	H
16	(	46	I
17	)	47	J
18	>	48	K
19	<	49	L
20	=	50	M
21	+	51	N
22	−	52	O
23	*	53	P
24	/	54	Q
25	;	55	R
26	,	56	S
27	.	57	T
28	0	58	U
29	1	59	V
30	2	60	W
31	3	61	X
32	4	62	Y
33	5	63	Z
34	6		
35	7		
36	8		
37	9		
38	A		
39	B		
40	C		

Tabela de tamanhos de tela

	Número máximo de caracteres na horizontal (ou número de colunas)	Número máximo de caracteres na vertical (ou número de linhas)
TRS-80	64	16
TK-82/TK-83	32	22
TK-85	32	22
Apple	40	25

Tabela de equivalência

Esta tabela mostra algumas das variações nas instruções de BASIC usadas pelos micros mencionados neste livro. Não foram incluídas as instruções para gerar gráficos, sons e cores, pois variam de fabricante para fabricante. Observe também que embora quase todos os micros usem as instruções PEEK e POKE, os endereços da memória variam de acordo com o fabricante, de modo que os números usados com essas instruções variam de computador para computador.

	Apple	TRS-80	TK-85	TK-82/TK-83
Escolha um número ao acaso entre 0 e 0,99999999	RND(1)	RND(0)	RND	RND
Escolha um número ao acaso entre 1 e N	RND(1)*N+1	RND(N)	RND*N+1	RND*N+1
Escolha uma letra ao acaso entre A e Z	CHR$(INT(RND(1)*26+65))	CHR$(RND(26)+64)	CHR$(INT(RND*26+65))	CHR$(INT(RND*26+38))
Limpe a tela	HOME	CLS	CLS	CLS
Verifique a tecla que está sendo apertada	X$=" " IF PEEK(−16384) > 127 THEN GET X$	INKEY$	INKEY$	INKEY$
Transforme caracteres em número de código	ASC ("X") (código ASCII)	ASC ("X") (código ASCII)	CODE ("X") (código ASCII)	CODE ("X") (código TK-82C)
Mova o cursor para cima	CALL − 998	PRINT CHR$(27)	PRINT CHR$(11)	PRINT CHR$(112)
Mova o cursor para baixo	PRINT CHR$(10)	PRINT CHR$(26)	PRINT CHR$(10)	PRINT CHR$(113)
Mova o cursor para a esquerda	PRINT CHR$(8)	PRINT CHR$(24)	PRINT CHR$(8)	PRINT CHR$(114)
Mova o cursor para a direita	PRINT CHR$(21)	PRINT CHR$(25)	PRINT CHR$(9)	PRINT CHR$(115)
Separe os primeiros N caracteres de um "string"	LEFT$(A$,N)	LEFT$(A$,N)	A$(1 TO N)	A$(1 TO N)
Separe os últimos N caracteres de um "string"	RIGHT$(A$,N)	RIGHT$(A$,N)	A$(N TO)	A$(N TO)
Separe N caracteres do meio de um "string"	MID$(A$,N1,N2)	MID$(A$,N1,N2)	A$(N1 TO N2)	A$(N1 TO N2)

Respostas

Pode ser que as suas respostas de alguns problemas sejam diferentes das que aparecem aqui. Isto não importa, contanto que funcionem no seu computador. Entretanto, verifique se são tão simples e concisas como as respostas do livro.

Página 5
Decolagem

Os números que determinam qual será a força são escolhidos nas linhas 30 e 40. Para mudar a máxima força possível, você pode mudar o 20 na linha 30, o 40 na linha 40 ou os dois números ao mesmo tempo. É claro que se você aumentar a máxima força possível o jogo ficará mais difícil.

Página 7
Jogos Intergalácticos

Mude as linhas 222 e 230 para:

```
222 LET B=B+INT(1000/G)
230 GOTO 20
```

e acrescente uma nova linha 15:

```
15 LET B=0
```

Página 9
Pirata Cósmico

Mude as linhas 30 e 40 e acrescente uma nova linha 35:

```
30 PRINT "DIFICULDADE? (6 A 30)"
35 INPUT S
40 LET G=INT(S/3)
```

Página 11
Os Olhos dos Monstros

Para fazer os monstros aparecerem em mais de quatro lugares na tela, você precisa colocar um número maior que 4 no meio da linha 70, mudar a linha 80 e colocar novas sub-rotinas no final do programa, uma para cada posição adicional.

Aqui estão as modificações para fazer com que os monstros apareçam em 5 lugares:

```
   70 LET R=INT(RND*5+1)
TK 80 GOSUB 220+20*R
■● 80 ON R GOSUB 240,260,280,300,320
  240 LET D=5
  245 LET A=1
  250 GOTO 350
  260 LET D=1
  265 LET A=9
  270 GOTO 350
  280 LET D=5
  285 LET A=18
  290 GOTO 350
  300 LET D=10
  305 LET A=7
  310 GOTO 350
  320 LET D=15
  325 LET A=15
  330 GOTO 350
```

Você pode colocar os números que quiser em A e D, contanto que caibam na sua tela.

Para aumentar o número de monstros, mude o 10 nas linhas 30 e 220 para um número maior. (Use o mesmo número nas duas linhas.)

Página 13
Módulo Lunar

Para aumentar a velocidade máxima permitida para um pouso seguro, você precisa mudar as linhas 240, 250 e 260. Você pode usar os números que quiser; quanto maiores, mais fácil será o jogo. No exemplo abaixo, você precisa de uma velocidade de 2 ou menos para uma boa aterrissagem e uma velocidade de 7 ou menos para uma aterrissagem razoável:

```
240 IF V1>7 THEN PRINT "VOCES
    CAIRAM...TODOS MORTOS"
250 IF V1>2 AND V1<=7 THEN PRINT "OK
    ...MAS ALGUNS FERIDOS"
260 IF V1<=2 THEN PRINT "BOA ATERRISSAGEM"
```

Página 15
Feras de Galacticon

Aqui estão quatro maneiras de tornar o jogo mais difícil:

1. Começar o jogo com um número menor de pessoas, colocando na linha 40 um número menor que 5.

2. Aumentar o número de feras, mudando o número 4 nas linhas 20 e 30. Colocar os nomes das novas feras das linhas 81 a 89, usando M$ (5), M$ (6), etc.
3. Aumentar o número de ataques, mudando o 8 na linha 160.
4. Aumentar a probabilidade de que a fera fique enfurecida, mudando o .4 na linha 330 por um número maior.

Página 17
Emboscada

Neste jogo, N é o número de código. Para fazer com que a contagem corresponda ao número de código, você precisa fazer com que a contagem aumente de N em vez de aumentar de 1. Para isso, substitua a linha 190 por:

```
TK 190 IF I$=CHR$(CODE(L$)+N) THEN
       LET S=S+N
■● 190 IF I$=CHR$(ASC(L$)+N) THEN
       LET S=S+N
```

Página 19
Cinturão de Asteróides

Você precisa mudar a linha 260 para que o computador some à contagem o número de estrelas em lugar de 1. O número de estrelas é controlado pelo valor escolhido para N na linha 70; assim, como no jogo acima, você precisa somar N à contagem. Você também terá que mudar a linha 320.

```
260 LET S=S+N
320 PRINT "VOCE FEZ ";S;" PONTOS"
```

Página 21
Viagem ao Futuro

1. Para aumentar o número de anos que devem passar na Terra, mude o 100 na linha 30 para um número maior. O exemplo abaixo é para 150:

```
30 LET T=INT(RND*150+25)
```

2. Para diminuir a tolerância de 5 para 2 anos, mude os números 5 nas linhas 180, 190 e 200 para 2:

```
180 IF ABS(T-T2)<=2 THEN PRINT "VOCE
    CHEGOU NA HORA"
190 IF (T-T2)>2 THEN PRINT "VOCE
    CHEGOU MUITO CEDO"
200 IF (T-T2)<-2 THEN PRINT "VOCE
    CHEGOU MUITO TARDE"
```

3. A linha 170 contém o número que determina o seu tempo de vida. Para aumentar (diminuir) o tempo de vida, basta substituir 50 por um número maior (menor).

Página 23
Vale da Morte

Você pode tornar o vale mais comprido substituindo o número 200 na linha 30 por um número maior.

Página 25
As Minas de Astron

Para o computador lhe perguntar se quer jogar de novo, acrescente as linhas abaixo:

```
645 PRINT ,"JOGA DE NOVO? (S OU N)"
646 INPUT A$
647 IF A$="S" THEN GOTO 10
```

Para somar a quantia com que terminou o jogo à quantia inicial para o jogo seguinte, introduza uma nova linha 5 e modifique a linha 30:

```
5 LET M=0
30 LET M=M+INT(RND*50+10)*P
```

(Use a versão correta de RND para o seu micro.)

Título original inglês
COMPUTER SPACEGAMES

Impresso no Brasil
ISBN 85-19-23219-1

PEDIDOS PELO REEMBOLSO POSTAL
Caixa Postal 23.052 — Rio de Janeiro, RJ — 20922